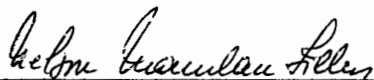


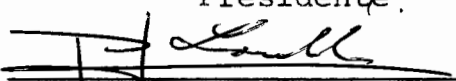
"SISTEMA DE EXIBIÇÃO GRÁFICA NÃO ITERATIVO"

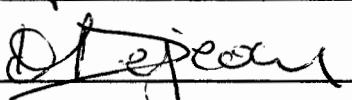
Antonio Lacerda Lima

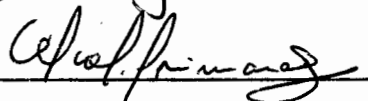
TESE SUBMETIDA AO CORPO DOCENTE DA COORDENAÇÃO DOS PROGRAMAS DE PÓS-GRADUAÇÃO DE ENGENHARIA DA UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE MESTRE EM CIÊNCIA (M.Sc.)

Aprovada por:



Presidente.






RIO DE JANEIRO
ESTADO DA GUANABARA - BRASIL
MARÇO DE 1974

À Josette

AGRADECIMENTOS

- Ao Professor François Gallais-Hamonno, pela orientação do início desta tese.

- Ao Professor Nelson Maculan Filho, que orientou este trabalho na sua fase final, leu e corrigiu o original.

- Ao Professor Pierre-Jean Lavelle, pela ajuda que me prestou, principalmente nos problemas concernentes à operação do sistema de computação MITRA 15.

- À COPPE, à Universidade Federal do Pará e a todas as Entidades que direta ou indiretamente contribuíram para a execução deste trabalho.

- A meus pais, em especial, por tudo que fui e que sou hoje.

RESUMO

Descrição geral de sistemas de exibição gráfica conversacionais.

Noções gerais sobre o sistema de computação MITRA 15 da Compagnie Internationale pour l'Informatique - CII.

Exposição do "software" de um sistema de exibição gráfica não iterativo destinado à geração de figuras geométricas planas.

ABSTRACT

A general description of interactive grafic exhibition systems.

General notions about the MITRA-15 system of the "Compagnie Internationale pour l'Informatique"- CII.

Exposition of the software of a noninteractive grafic exhibition system used for plane geometric pictures generat'on.

ÍNDICE

<u>CAPITULO I</u>	pg.
Introdução	1
<u>CAPITULO II</u>	
Sistemas Gráficos-Display	
2.1 - Descrição geral	4
2.2 - Geração da Imagem	8
2.2.1 - Geração da Imagem por Pontos	8
2.2.2 - Geração da Imagem por Vetores	11
2.2.3 - Geração de Caracteres	13
2.3 - Dispositivos de Entrada	
2.3.1 - Light Pen	13
2.3.2 - Quadro de Funções	15
2.4 - Controle das Interrupções	16
<u>CAPITULO III</u>	
Considerações sobre o MITRA-15	18
3.1 - Memória Principal	18
3.2 - Unidade de Tratamento	18
3.3 - Estrutura de um Programa	21
3.4 - Elementos de Comunicação de um Programa	
3.4.1 - Bloco de Trabalho (TWA)	23
3.4.2 - Tabela de Afetação das Seções (PRT)	24
3.4.3 - Tabela das Palavras de Desativação (DVT)	24
3.4.4 - Contexto (CTX)	25
3.5 - Gestão das Zonas de Dados	
3.5.1 - Zona de Dados Comuns ao Sistema (ZC)	26
3.5.2 - Zona de Dados Locais	26
<u>CAPITULO IV</u>	
Sistema de Exibição Gráfica-SEG	
4.1 - Objetivos	28
4.2 - Descrição Geral do Sistema	28
<u>CAPITULO V</u>	
Estrutura dos dados	
5.1 - Estrutura do Usuário	32
	36

5.2 - Estrutura do Display	36
5.2.1 - Informações de Controle	37
5.2.2 - Informações a serem transmitidas ao Display	39
5.3 - Organização da Estrutura do Display	42

CAPITULO VI

Geração de Figuras	44
6.1 - Subrotinas Básicas	44
6.1.1 - DEFIG	45
6.1.2 - FIM	46
6.1.3 - LIGSF	47
6.1.4 - GEFIG	48
6.1.5 - CINT1	49
6.1.6 - CIRCO	51
6.1.7 - CIRC	53
6.1.8 - CRIMD	55
6.1.9 - GEMEN	56
6.1.10- LPCEI	58
6.2 - Programa de Controle e Exibição da Imagem	59
6.2.1 - Subrotina EXEC	64

<u>CONCLUSÃO</u>	66
------------------	----

APENDICE :

Listagem da Programação dos Algoritmos	
DEFIG	69
FIM	70
GEFIG	71
LIGSF	73
CINT1	74
CIRCO	76
CRIMD	77
CIRC	79
GEMEN	80
LPCEI	82
Prog. Contr. Exib. Imagem	83

<u>BIBLIOGRAFIA</u>	87
---------------------	----

CAPÍTULO 1

INTRODUÇÃO

Atualmente a computação gráfica não constitui apenas um campo de especulações e ensaios em laboratórios; as pesquisas desenvolvidas nas duas últimas décadas, transformou-a numa poderosa técnica que se identifica com um vasto campo de atividades, tais como: projetos de circuitos eletrônicos, engenharia civil, arquitetura, engenharia nuclear, aeronáutica, medicina, etc.

Os esquemas de entrada e saída dos consoles gráficos, associados aos modernos conceitos de Linguagens de Programação, habilitam os computadores a uma fácil e fluente interação com o elemento humano.

A conexão de tubos de raio catódicos, semelhantes aos usados em televisão, aos computadores não só possibilitaram a visualização de informações nestes armazenadas e processadas, como também, utilizando a Light Pen, funcionam como entrada de dados em forma de figuras e fornecem imediatas respostas.

Esta interação , homem-máquina, apresenta grandes vantagens:

- a) O usuário pode imediatamente ver qualquer erro grosseiro nas suas figuras geradas e corrigi-los usando Light Pen ou alterando os comandos do seu programa;
- b) Ele pode acompanhar o progresso da solução de um dado problema , tomando as decisões que se façam necessárias;
- c) O "display" gráfico pode apresentar dados cuja interpretação e

entendimento tornar-se-iam difíceis se a emissão deste fosse via uma impressora ou um plotter. Com uma hábil programação, o usuário pode gerar várias vistas de um dado objeto; movimentar as figuras; gerar linhas cintilantes, interrompidas e de intensidades diferentes.

O alto custo associado à implementação destas técnicas tornam as mesmas indisponíveis às pequenas e médias empresas, ficando assim a utilização restrita às grandes empresas e às universidades.

Destacaremos a seguir alguns dos atuais empregos dos consoles gráficos:

1) Diagnósticos médicos - Um console gráfico para fins médicos necessitará de detalhadas informações referentes à história médica do paciente que, quando requeridas, estarão em poucos segundos, visualizadas sobre um vídeo.

2) Construção de modelos moleculares - Modelos de estruturas moleculares são preparados- o arranjo dos átomos na molécula - e armazenados na memória do computador para posteriores visualizações. Se desejado, utilizando rotações, o modelo pode ser visualizado em suas três dimensões.

3) Simulação de sistemas discretos - O modelo de uma determinada situação em análise é fornecido ao computador. Um exemplo clássico é a pesquisa em um sistema de filas onde existem linhas de espera de objetos ou processos. Imagens destes modelos são geradas sobre um vídeo, tal que a qualquer variação dos parâmetros ou características do modelo, as alterações ocorridas são prontamente visualizadas. Em sistemas iterativos, os modelos podem ser construídos u-

tilizando-se o Light Pen.

4) Simulação de sistemas contínuos - Estes sistemas podem ser construídos, simulados, visualizados e estudados de modo análogo aos sistemas discretos.

5) Edição de texto - Utilizando Light Pen ou outros tipos de entrada de dados, pode ser adicionado, eliminado e modificado o conteúdo de um texto visualizado sobre o vídeo.

6) Administração - Bancos de dados contendo vários tipos de informações comerciais podem ser utilizados para gerar mapas ou gráficos que traduzem determinadas situações, possibilitando assim, rápidas e seguras tomadas de decisões.

CAPÍTULO 2

SISTEMAS GRÁFICOS - DISPLAY

2.1 - DESCRIÇÃO GERAL

Um típico sistema gráfico consiste de vários consoles gráficos, compartilhando o tempo de um processador central (CPU). Se os consoles liberam o computador, este pode processar em batch e em menor prioridade (caso possua) executar alguma função do quadro de funções (keyboard).

A figura 1 mostra um sistema dedicado à computação gráfica, constituído de um computador e um console gráfico.

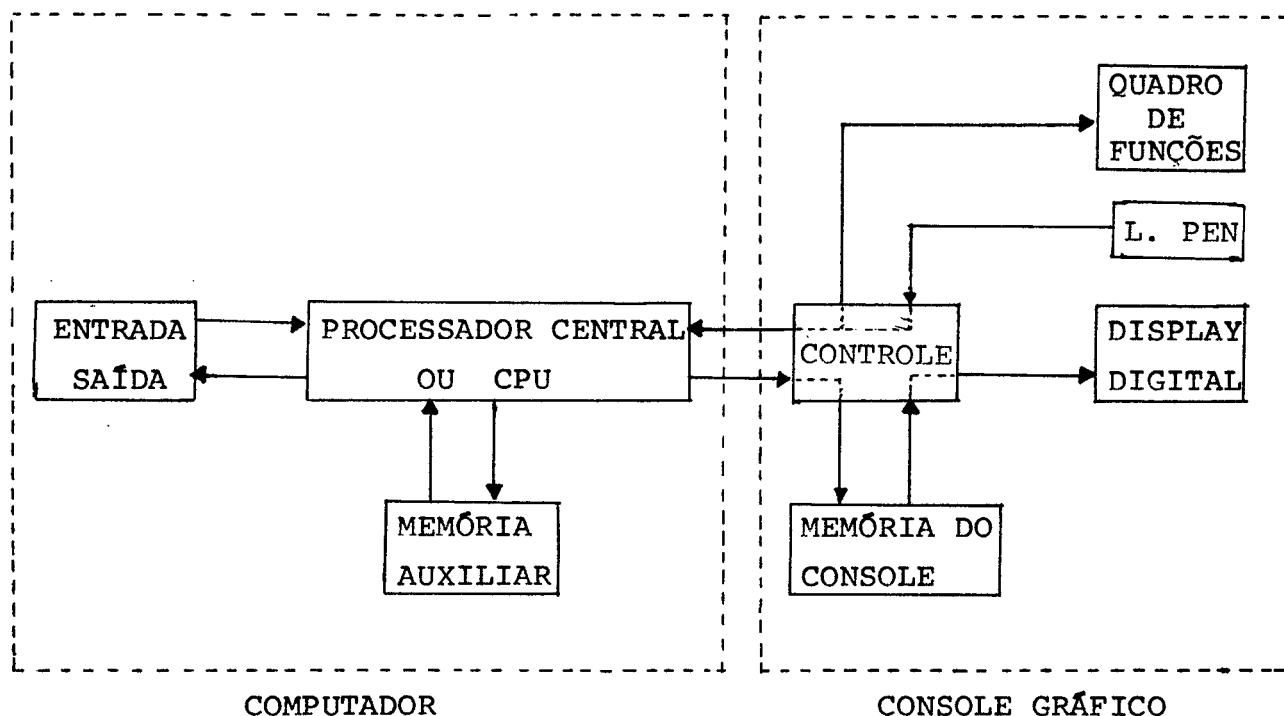


Fig. 1

O console contém uma área de memória própria, um controle e um display digital. A realimentação do computador faz-se

via o Light pen, Keyboard ou outros dispositivos de que disponha o console; o controle identifica a natureza e procedência das interrupções e transmitirá à CPU as ordens a serem executadas. Na memória do console é armazenada uma sequência de palavras - DISPLAY LIST - que define a figura a ser traçada sobre o vídeo. A persistência da figura sobre o vídeo exige uma contínua reestauração da imagem ; isto fica ao cargo do controle que lê o conteúdo da memória do console e o transmite ao display. Quando a lista do display tem que ser gerada ou modificada, o controle recebe os dados transmitidos pela CPU e gera ou atualiza a lista do display.

A memória central do computador é dividida em três seções, conforme mostra a Fig. 2 .

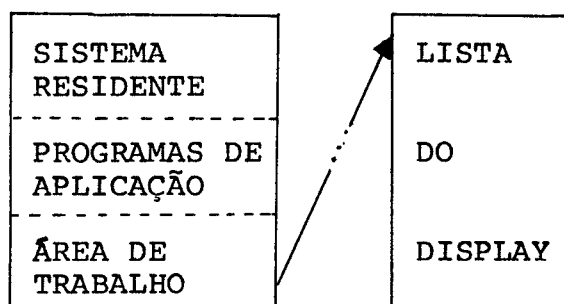


Fig. 2

Os dados e informações geradas na área de trabalho são transferidos à memória do console sob a supervisão do sistema operacional e do controle do console.

No sistema apresentado, uma vez gerada a lista do display, a CPU está apta a processar novos dados, quer sejam referentes a mudanças na lista do display ou execução de programas que estavam

ã espera sem que isto venha a prejudicar o bom desempenho do sistema, tendo em vista que a geração e restauração da imagem fica a cargo do controle do console.

Contudo, nem sempre é este o método utilizado; existem sistemas menos sofisticados em que o controle do display é exercido pela própria CPU e uma parte da memória central é destinada a armazenagem da lista do display. Vide Fig. 3 .

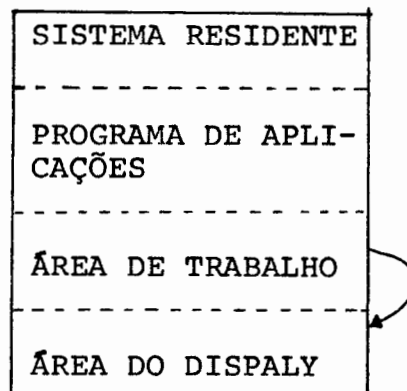


Fig. 3

Neste caso, um programa residente percorre a lista do display e transfere seu conteúdo (uma palavra de cada vez) diretamente aos registros do display. Ao atingir o fim da lista do display, testa se chegou algum sinal de interrupção; se não, reinicia a restauração da imagem. Caso contrário, é identificada a origem da interrupção, se proveniente do Light Pen, do quadro de funções ou outros dispositivos. Executa as ações requeridas para a solução do problema proposto ou a modificação da figura e retorna à restauração da imagem. O tempo gasto para completar um ciclo, var-

redução da lista de display e atendimento de uma interrupção, dependerá do tamanho da lista e da computação da ordem recebida. Se este tempo ultrapassar o período de excitação do tubo de raios catódicos-CRT, a figura se apresenta bruxuleante ou mesmo pode se apagar, até que a restauração seja reiniciada.

Alterar ou substituir uma figura que está sendo exibida requer a computação de novas ordens e a geração de uma nova lista do display, superpondo-se à anterior. Tendo em vista o acima citado, isto pode acarretar sérios problemas no desempenho do sistema. Uma solução viável é o programa que gerará a nova lista do display compartilhar com o controle do display o tempo da CPU. Após cada ordem transferida ao display existe um tempo de espera exigido para a execução da mesma; nestes pequenos intervalos será construída a nova lista do display na área de trabalho; a atualização do ponteiro da área do display transferirá o processamento para a nova lista, a mudança da imagem sob o vídeo será imediata e sem prejuízo ao bom funcionamento do sistema. A comutação entre os dois programas pode ser controlada pela prioridade do que controla o display; este será lançado sempre que o display emitir um sinal de que a ordem recebida foi executada. Uma segunda alternativa é usar um relógio interno que em intervalos pré-determinados gera uma interrupção sobre o programa em execução e lança o programa de controle. Após o envio de nova ordem ao display, o programa de controle será desativado e isto automaticamente lançará o programa que se encontra interrompido.

2.2 - GERAÇÃO DE IMAGEM

A complexidade de uma figura que deve ser exibida pelo display é limitada pelo bruxuleamento da figura e pelo tamanho da memória reservada para a lista do display. Estes dois fatores dependem da técnica usada no deslocamento do raio luminoso sob o vídeo. Uma técnica simples, requer um excessivo tempo de computação, muita memória e o deslocamento do raio é vagaroso. Quanto mais funcional é a técnica usada, mais complexos são os equipamentos utilizados; contudo, o tempo de computação e a memória ocupada, são substancialmente reduzidos, o deslocamento do raio luminoso é rápido. Estes fatores asseguram a manipulação de figuras complexas.

2.2.1 - GERAÇÃO DE IMAGEM POR PONTOS

Nesta técnica, cada palavra da lista do display representa um ponto a ser visualizado sobre o vídeo, de um modo geral cada palavra de N bits e subdividida em 4 campos:

- a) Intensidade (luminosa ou não) ; 1 bit
- b) Coordenada X ; $(N-2)/2$ bits
- c) Coordenada Y ; $(N-2)/2$ bits
- d) Escapa ; 1 bit

O campo d só é necessário se o display opera em mais de um modo. Exemplos: ponto, vetor, caracter, etc . Se o bit de escape é zero, o sistema assume que o modo de operação não será alterado para a próxima palavra. Se o bit é 1 , a próxima palavra indicará o novo modo de operação do sistema.

A Fig. 2 mostra o esquema básico de um display digital gerador de pontos.

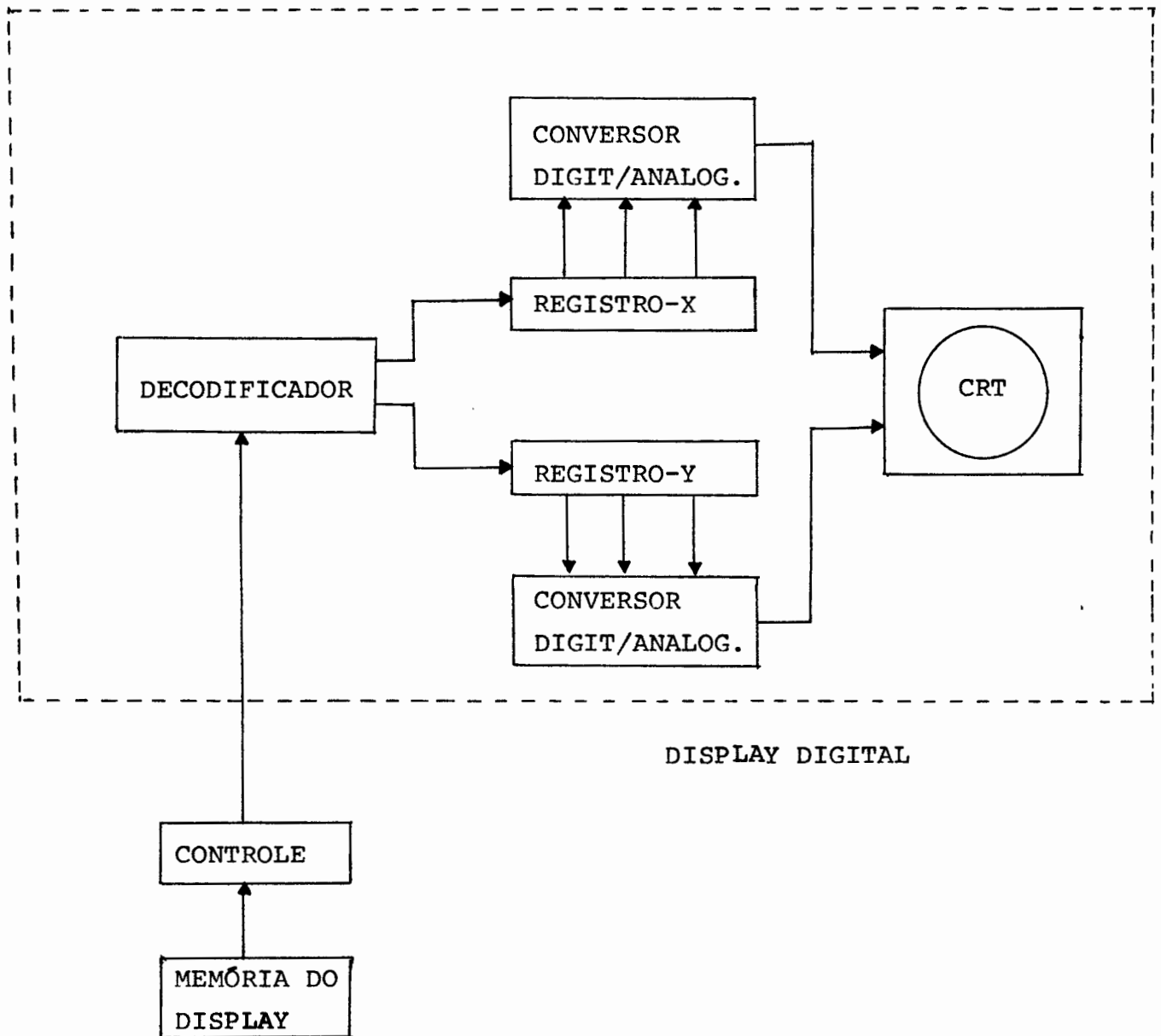


Figura 2

A informação transferida ao display digital pelo controle é recebida pelo decodificador que separa os campos acima citados; as coordenadas são armazenadas em seus respectivos registros; conversores digital/analógicos conectados aos registros deduzem , a partir dos valores das coordenadas, a voltagem que deflektará o raio luminoso sobre o CRT . Após a deflecção, o decodificador manda ao CRT o sinal da intensidade do ponto que está sendo gerado e então novas coordenadas podem ser transferidas aos registros X e Y .

As vantagens deste método são a simplicidade dos circuitos e a inexistência de acumulação de erros na geração da imagem, visto que as coordenadas são expressas em valores absolutos.

As desvantagens são:

- a) Gasto excessivo de memória, visto que cada ponto será especificado por uma palavra da memória e as linhas retas e curvas serão geradas por sequência de pontos, em média 45 pontos por centímetro.
- b) A sobrecarga do computador, uma vez que cada ponto tem que ser processado individualmente.
- c) Pode ser requerido tempo demais na exibição de uma figura.
- d) O intervalo que separa a geração de dois pontos é fixo e tomado por base a separação máxima entre eles.

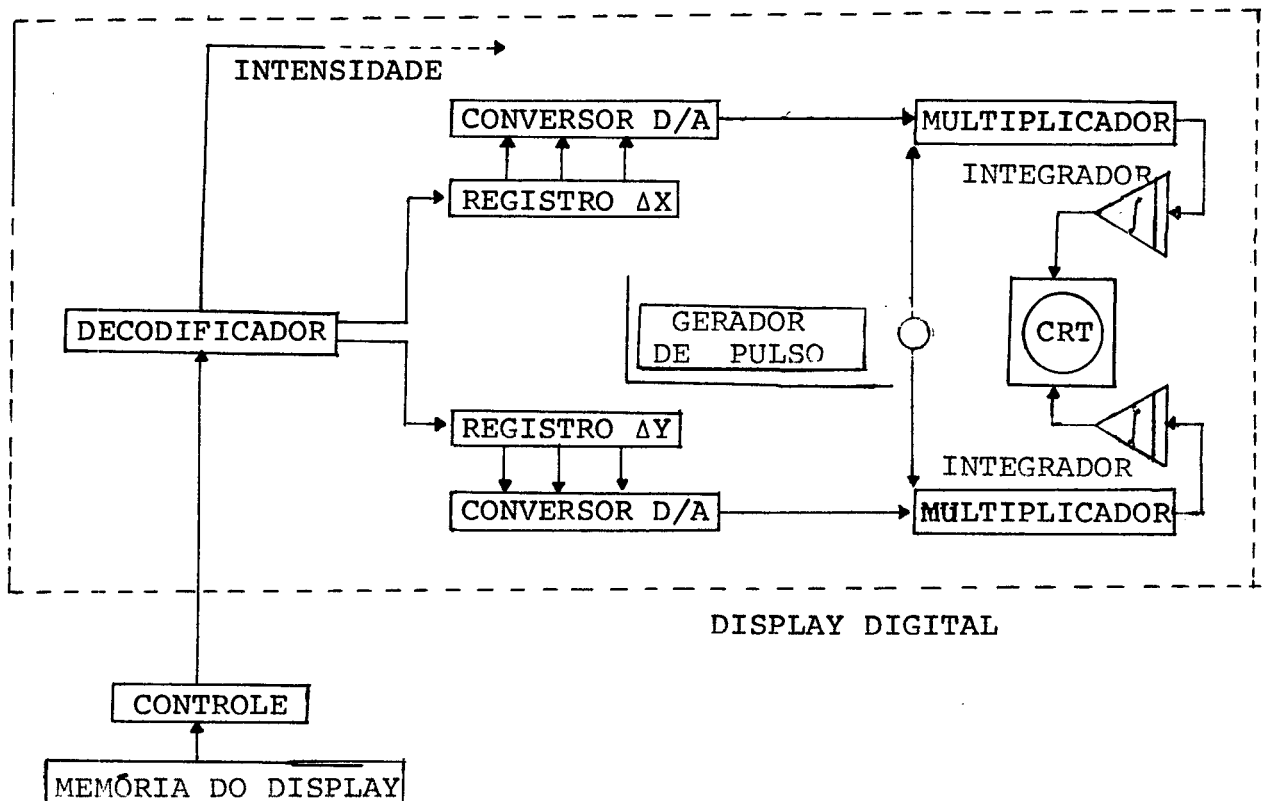
Em síntese, a geração de imagem por pontos é conveniente quando pontos esparsos têm que ser gerados; contudo, é muito ineficiente e anti-econômica quando figuras a base de linhas retas e curvas são manipuladas.

2.2.2 - GERAÇÃO DE IMAGEM POR VETORES

Nesta técnica, cada palavra da lista do display especifica o deslocamento vertical e horizontal do raio luminoso a partir de uma posição (X,Y), onde este se encontra sobre o vídeo. Cada ordem transferida ao display é composta de 6 campos. Suponhamos que cada ordem constitui uma palavra de $2N$ bits. Sua estrutura seria:

- a) Intensidade (luminosa ou não) ; 1 bit
- b) Sinal de ΔX ; 1 bit
- c) Valor de ΔX ; $(N-2)$ bits
- d) Sinal de ΔY ; 1 bit
- e) Valor de ΔY ; $(N-2)$ bits
- f) Escape ; 1 bit

A Figura 3 mostra o esquema de um display digital gerador de vetor.



Os valores digitais contidos nos registros ΔX e ΔY são convertidos a valores analógicos e recebidos pelos multiplicadores que, sob a ação de um gerador de pulsos sincronizados, emitem uma sequência de valores analógicos ao integrador analógico, cuja saída determina o posicionamento do raio luminoso sobre o vídeo. Evidentemente a função do multiplicador e integrador é substituir o conjunto de palavras que seriam usadas para definir uma linha se a geração por pontos fosse utilizada, com a vantagem de que todos os pontos gerados por este processo são adjacentes e portanto mais rápidos.

Embora esta técnica seja muito rápida e pouco dispendiosa, ela apresenta algumas dificuldades, tais como:

- a) O brilho dos vetores traçados variam com o seu comprimento, a não ser que uma compensação na intensidade seja usada.
- b) A saída do integrador analógico está sujeita a flutuações que podem causar deformações ou não fechamento nas figuras geradas; isto é possível quando uma série de vetores é operada sequencialmente.

Entre as vantagens apresentadas por este método, podemos destacar:

- a) Gasto reduzido de memória.
- b) Elimina a computação que envolve o processamento de pontos individuais.
- c) Alta velocidade na geração das figuras.
- d) Facilita a descrição e programação das figuras que, uma vez processadas, podem ser exibidas a partir de qualquer ponto do vídeo.

2.2.3 - GERAÇÃO DE CARACTERES

Os dados alfanuméricos a serem manipulados consistem de um conjunto de letras, números e caracteres especiais, aptos a representarem qualquer informação alfanumérica.

A reprodução dos caracteres sobre o vídeo obedece a regras pré-estabelecidas; é impossível a visualização dos caracteres em ângulos e tamanhos arbitrários. Caracteres de tamanhos diferentes são gerados por conjuntos distintos.

A geração de caracteres pode ser ao nível de hardware ou software. Em ambos os casos o conjunto de caracteres é representado por uma tabela de códigos. Em nível de software identifica uma sub-figura que descreve um dado caracter; uma mensagem será constituída de uma série de desvios da figura principal para as sub-figuras que gerarão a mensagem em apreço. A localização dos caracteres sobre o vídeo dependerá exclusivamente da posição do raio luminoso quando a sub-figura é chamada. Em nível de hardware, os códigos são transmitidos ao display que automaticamente gera o caracter; deste modo, o computador se libera da manipulação das sub-figuras e economiza-se a memória que deveria ser ocupada na descrição dos caracteres.

2.3 - DISPOSITIVOS DE ENTRADA

2.3.1 - LIGHT PEN

A light pen é o mais importante dos instrumentos de

realimentação de um sistema gráfico. Ela é usada para detectar pontos, linhas ou caracteres sobre o vídeo. Quando utilizada em conjunto com o quadro de funções programadas, ela pode ser usada para produzir construções ou modificações de gráficos complexos.

A light pen, Figura 6, consiste de um feixe de fibra ótica que conduz a luz do vídeo a um tubo fotomultiplicador.

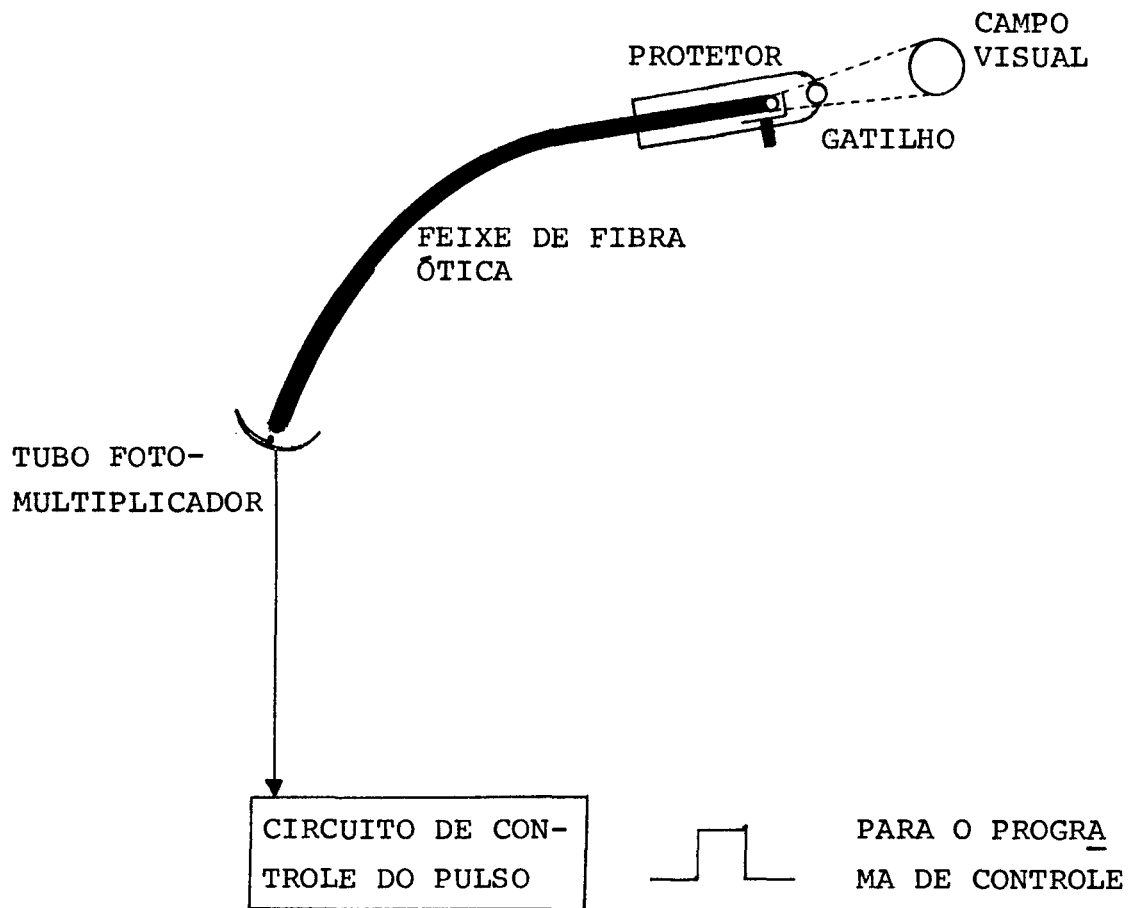


Figura 6

Quando um ponto ou vetor sobre o vídeo está dentro do campo visual da light pen, um pulso é gerado pelo fotomultipl*ic*ador cada vez que o ponto ou vetor é restaurado. O pulso é rece*bi*do pelo controle que o transmite ao computador; a simultaneida-

de, considerado o tempo de atraso na programação do pulso, entre a restauração de um dado elemento da figura e a geração do pulso permite a identificação deste elemento.

Um protetor na extremidade do feixe da fibra ótica facilita o manuseio deste. Um anteparo na extremidade do protetor evita que a light pen seja excitada, enquanto um gatilho que movimenta o anteparo não seja acionado.

2.3.2 - QUADRO DE FUNÇÕES PROGRAMADAS

Ele é usado para requisitar funções previamente programadas e que fazem parte do sistema residente. Cada tecla do keyboard é associada a uma função; a compressão de uma tecla, um sinal de interrupção e o código da função são mandados ao programa de controle. A função requerida é pesquisada e lançada com os dados que normalmente são fornecidos através do light pen.

Quando um grande número de funções é codificado, é conveniente o uso das chaves especiais que identificarão um dado conjunto de funções.

A unidade de display 2250 da IBM utiliza um quadro de funções com 32 teclas e uma tabela de overlay em sua extremidade superior. Cada função é identificada por sua chave, de 0 a 31, e por seu código de overlay, de 0 a 7 . Deste modo, 256 funções podem ser catalogadas.

2.4 - CONTROLE DAS INTERRUPÇÕES

Um sistema conversacional requer equipamentos e programas que o habilite a responder rapidamente as perguntas que lhe são fornecidas. A eficiência deste processamento em tempo real depende em grande parte das facilidades apresentadas pelo programa que controla as interrupções.

O fato das operações de entrada e saída serem executadas por subrotinas residentes, libera o usuário dos detalhes do seu funcionamento. Contudo, torna-se necessário a compreensão destas subrotinas a fim de que se possa entender os métodos de comunicação entre o computador e o console gráfico.

Quando um periférico requisita os serviços do processador central, um sinal desta unidade suspende o programa que está em execução e o processamento é desviado para a primeira instrução da subrotina de interrupção. Após a execução desta subrotina, o processamento retorna ao programa principal, reiniciando da primeira instrução que teria sido executada se a interrupção não tivesse ocorrido.

Quando vários periféricos podem causar interrupções, torna-se ineficiente o tratamento de todas as interrupções por uma única subrotina. Além do mais, certos periféricos requerem uma resposta mais rápida que outros e pode ainda ocorrer que mais de um periférico requisi-te os serviços do processador central ao mesmo tempo. Em vista disto, é conveniente que o sistema identifique e atenda às interrupções de maior importância.

Este problema é solucionado distribuindo-se as unidades externas em níveis de execução diferentes, cuja prioridade é em função de sua importância. Uma interrupção sob um dado nível causa o desvio do processamento para a subrotina relativa ao nível que interrompe. A interrupção só é atendida de imediato se ela parte de um nível de prioridade maior do que está em execução; caso contrário, ela ficará em espera até que nenhum outro nível de maior prioridade requisite os serviços do processador central.

Se mais de um periférico utiliza um mesmo nível, então o posicionamento de determinados bits na palavra de interrupção daquele nível identificará o periférico que deve ser atendido. Esta identificação é a primeira providência a ser tomada pela subrotina de interrupção.

CAPÍTULO 3

CONSIDERAÇÕES SOBRE O MITRA 15

O MITRA 15 é um mini-computador construído pela Com
pagnie Internationale pour l'Informatique - CII - à base de circuit
os integrados.

Seu processamento em tempo real, sua concepção modu
lar e sua estrutura microprogramada muito moderna, permite-lhe a-
bordar com grande eficácia, um grande número de aplicações: tele-
informações, processos industriais, cálculo científico, etc.

3.1 - MEMÓRIA PRINCIPAL

A memória principal do MITRA 15 é organizada em pal
avras de 16 bits , mais 1 bit de paridade , e 1 bit de proteção
(total = 18 bits) . O acesso muito rápido é de 800 nano-segundos
por palavra, ou seja, 1,25 milhões de palavras por segundo.

A memória pode variar de 1 a 8 blocos, contendo cada
bloco 4.096 palavras. Atualmente dispomos de 4 blocos, ou seja,
16.384 palavras.

A memória é endereçável por octetos e alterável por
octeto, palavra ou palavra dupla.

3.2 - UNIDADE DE TRATAMENTO

A unidade de tratamento é composta de um bloco de reg

gistros rápidos, cinco indicadores do programa, uma memória morta microprogramada, um operador e um sistema de interrupção.

Segundo a microprogramação implantada em sua memória morta, uma unidade de tratamento pode efetuar as seguintes funções:

- Unidade central
- Unidade de carga
- Unidade especial adaptada a um tratamento particular.

A unidade de tratamento controla uma área da memória sob a qual estão conectados todos os seus periféricos.

Os registros são organizados em blocos de 8 registros de 16 bits. Cada unidade de tratamento pode conter 8 ou 16 blocos de registros. O tempo de acesso é de 60 nano-segundos por palavra registro.

O bloco 0, na unidade central, é destinado a armazenagem do contexto do programa em execução; os demais blocos são usados nas tarefas que envolvem os periféricos.

Estrutura do Bloc 0 :

- P - Contador do Programa
- L - Base Logal
- G - Base Geral
- A - Acumulador
- E - Extensão do Acumulador
- X - Indice
- V - Utilizado para microprogramação
- W - Utilizado para microprogramação

Todos os blocos de registros da unidade de carga são destinados às trocas com os periféricos.

A memória microprogramada , ROM , permite a leitura não destrutível , é realizada em circuitos integrados e cada palavra de 16 algarismos binários, compreende uma micro-instrução cujo tempo de execução é de 300 nanosegundos.

O formato da micro-instrução é o seguinte:

M	OP	CC	AD
---	----	----	----

M - Comando da memória

OP - Código principal da operação

CC - Código complementar

AD - Define o endereço da micro-instrução seguinte.

O sistema de interrupção é lançado quando:

- Um sinal de interrupção é gerado .
- Uma micro-instrução especial é executada.
- As interrupções não estão mascaradas.
- O nível do programa em curso é inferior ao da interrupção incidente.

Dos 32 níveis de interrupção do MITRA 15 , 4 são internos. Sobre os demais podem ser agrupadas 4 interrupções por nível, perfazendo um total de 112 interrupções externas.

O Endereço 10 (DEZ) da memória viva aponta para uma tabela de 32 palavras, cujo conteúdo aponta os contextos dos níveis de interrupção.

O registro 8 (oito) contém, multiplicado por 2, o número do nível cujo programa está sendo executado.

3.3 - ESTRUTURA DE UM PROGRAMA

A programação do MITRA 15 admite a modularidade dos seus programas. Cada módulo será designado pelo termo 'seção'.

As vantagens da modularidade são:

- Facilita a especificação de um sistema.
- Facilita sua realização em paralelo por vários programadores.
- Permite a reutilização de seções idênticas de um sistema pelo outro.
- Facilita o acerto e a continuação de um produto.

Uma seção ou módulo é constituída principalmente por uma sequência de interrupções-segmento de programa. Estas instruções podem se referenciar a dados locais ou a dados comuns a várias seções.

Os dados próprios a uma seção formam um segmento de dados locais - LDS. Os símbolos e etiquetas definidos no LDS só podem ser referenciados pelo segmento de programa - LPS e associado a esta, LDS. Uma sequência de LPS's pode ser associada a uma mesma LDS. O acesso aos dados pode ser direto, indireto ou indireto indexado.

Os dados comuns formam o segmento de dados comuns - CDS. Os símbolos e etiquetas definidos em um CDS são comuns a to

do o programa. O acesso destes dados, quer seja por meio do LDS ou LPS pode ser direto, indireto ou indireto indexado.

Do ponto de vista de hardware, a modularidade implica na existência de instruções especiais que executam a chamada ou o retorno de uma seção.

Quanto à programação, a modularidade dos programas é uma noção fundamental da linguagem assembler que comporta as diretivas ditas de seccionamento:

- CDS - segmento de dado comum
- LDS - segmento de dado local
- LPS - segmento de programa local
- FIN - fim de segmento ou seção
- IDS - segmento de dados indiretos

Cada segmento é terminado por uma diretiva 'fim' .

Um programa pode ser constituído por módulos de diversas origens, estes módulos são conectados via o editor de linhas para formar um programa completo.

Cada programa termina pela diretiva END .

Três apontadores principais controlam a execução de um programa . Estes são:

a) BASE GERAL - G

A base geral é associada de forma biunívoca ao programa; todos os endereços referenciados pelo programa lhe são relativos. Ela aponta para o primeiro endereço da CDS .

b) BASE LOCAL - L

A base L é uma base implícita associada a um segmento de dado local. Seu valor é relativo à base G e o endereço de todos os dados deste segmento lhe são relativos.

c) BASE PROGRAMA - P

Esta base é associada a um segmento de programa; inicialmente ela aponta à primeira instrução da seção ; quando a seção é lançada , ela toma um contador que aponta a próxima instrução a ser executada.

As bases L e P são definidas automaticamente pelo editor de linhas em valores relativos à base G , e são armazenadas na tabela de realocação de programa - PRT .

3,4 - ELEMENTOS DE COMUNICAÇÃO DE UM PROGRAMA

3.4.1 - BLOCO DE TRABALHO (TWA - TASK WORKING BLOCK)

Este bloco é constituído pelas primeiras 32 ou 16 palavras da CDS , caso seja utilizado um TWA extendido ou não. É reservado às atividades do monitor de base que controla as execuções dos programas.

No endereço G+6 , o monitor armazena o endereço relativo a G , da zona comum do sistema (ZC).

Quando o processamento é desviado para um dos módulos do supervisor, nesta área são armazenados o endereço de retorno

à base local (L) da tarefa chamante e os indicadores.

3.4.2 - TABELA DE AFETAÇÃO DAS SEÇÕES (PRT - PROG. RELOC. TABLE)

Uma afetação de seção é constituída por uma palavra dupla que contém os valores iniciais de P e L relativos a G:

$I = L - G$
$P = P - G$

Palavra dupla de afetação: (SRD - section Reloc. double-work)

A PRT é formada pelo conjunto das SRD das seções constitutivas de um programa. Ela precede imediatamente o endereço de G, de sorte que o endereço da SRD da seção n é igual a : $G - 4n$.

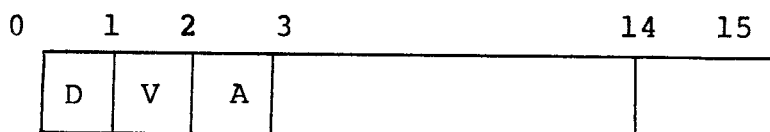
A PRT constitui o elemento de ligação entre as seções de um programa. A ligação entre o programa e o monitor é provida pela PRT do monitor; o endereço 12 da memória viva aponta para esta tabela.

A PRT de um programa é gerada na "linkagem" do programa.

3.4.3 - TABELA DAS PALAVRAS DE DESATIVAÇÃO - DVT

Ela precede na memória viva a tabela dos ponteiros dos contextos (CPT) e igualmente é formada por 32 palavras.

A estrutura de cada palavra da DVT é a seguinte:



Bit zero : Declaração da interrupção - D

Bit 1 : Validação da interrupção - V

Bit 2 : Armamento da interrupção - A

Bits 3 - 13 : Número do grupo da interrupção

Bits 14-15 : Grupo da interrupção

3.4.4 - CONTEXTO - CTX

O contexto é o elemento de ligação entre um nível de prioridade e o programa associado a este nível. Ele é composto de 7 palavras:

1^a palavra: Indicadores do estado do programa

2^a palavra: Valor inicial do registro X

3^a palavra: Valor inicial do registro E

4^a palavra: Valor inicial do registro A

5^a palavra: Valor inicial da base G

6^a palavra: Valor inicial da base L

7^a palavra: Valor inicial de P

O contexto permite a inicialização, o relançamento e a proteção do estado de uma tarefa, caso o nível correspondente a esta tarefa seja ativado ou desativado.

3.5 - GESTÃO DAS ZONAS DE DADOS

3.5.1 - ZONA DE DADOS COMUNS AO SISTEMA - ZC

Esta zona é acessível a todos os programas em memória, não importando o nível a que estejam associados. Seu endereço é armazenado pelo supervisor no endereço G+6 correspondente a cada programa. Sua localização no limite superior da memória resulta que todos os seus endereçamentos sejam positivos.

Ela é constituída por blocos de memória de tamanhos fixos que podem ser utilizados dinamicamente, conforme a demanda dos usuários.

Um programa acederá ao bloco através do endereçamento indireto geral indexado. A progressão no bloco é obtida pela incrementação ou decrementação do registro X, que inicialmente contém o endereço relativo a ZC do bloco.

3.5.2 - ZONAS DE DADOS LOCAIS

Vimos que cada seção ou módulo de um programa dispõe de um segmento de dados locais. Quando o processamento é desviado para uma dada seção, a atualização da base L se faz automaticamente e passa a apontar o segmento de dados locais daquela seção.

O formato das instruções do MITRA 15 só permite o acesso direto aos primeiros 256 bytes do LDS, isto é, ela dispõe de um campo de 8 bits, destinado ao deslocamento do dado a partir da base L. Contudo, caso seja necessário o acesso direto a dados fora deste limite, isto é possível pela incrementação ou decremen

tação da base L . O sistema dispõe de duas instruções que executam esta ação.

CAPÍTULO 4

SISTEMA DE EXIBIÇÃO GRÁFICA - SEG

4.1 - OBJETIVOS

Este trabalho constitui parte do projeto e implementação do software de base de uma unidade exibidora: DISPLAY.

Trataremos da geração e controle de figuras geométricas planas; a manipulação de figuras geométricas tridimensionais é assunto de outra tese que complementarará este trabalho.

As subrotinas básicas aqui descritas serão programadas em assembler - MITRA 15 e serão chamadas pelos programas dos usuários, escritos em linguagem Fortran.

O sistema ao nível de hardware somente gerará pontos, linhas retas e caracteres alfanuméricos. Todas as figuras devem ser compostas, ao nível de software, destes primitivos elementos.

O equipamento básico utilizado consiste de um osciloscópio conectado ao mini-computador MITRA 15.

4.2 - DESCRIÇÃO GERAL DO SISTEMA

No intuito de facilitar a descrição e o funcionamento básico do sistema, consideraremos que este é formado por um conjunto de blocos com funções específicas, mas que se interligam, formando um todo.

Designaremos por programa de aplicação, o programa fonte, em Fortran, fornecido pelo usuário. Os dados deste programa irão compor parte da estrutura dos dados do sistema.

Utilidades geométricas e parte de geração de imagem são conjuntos de módulos que, em vista da carência de memória auxiliar, comporão uma pequena biblioteca em fita de papel. Os módulos desta biblioteca serão integrados ao programa do usuário, na "link" edição deste, conforme suas necessidades.

SISTEMA GRÁFICO NÃO ITERATIVO P/ MITRA 15: DISPLAY

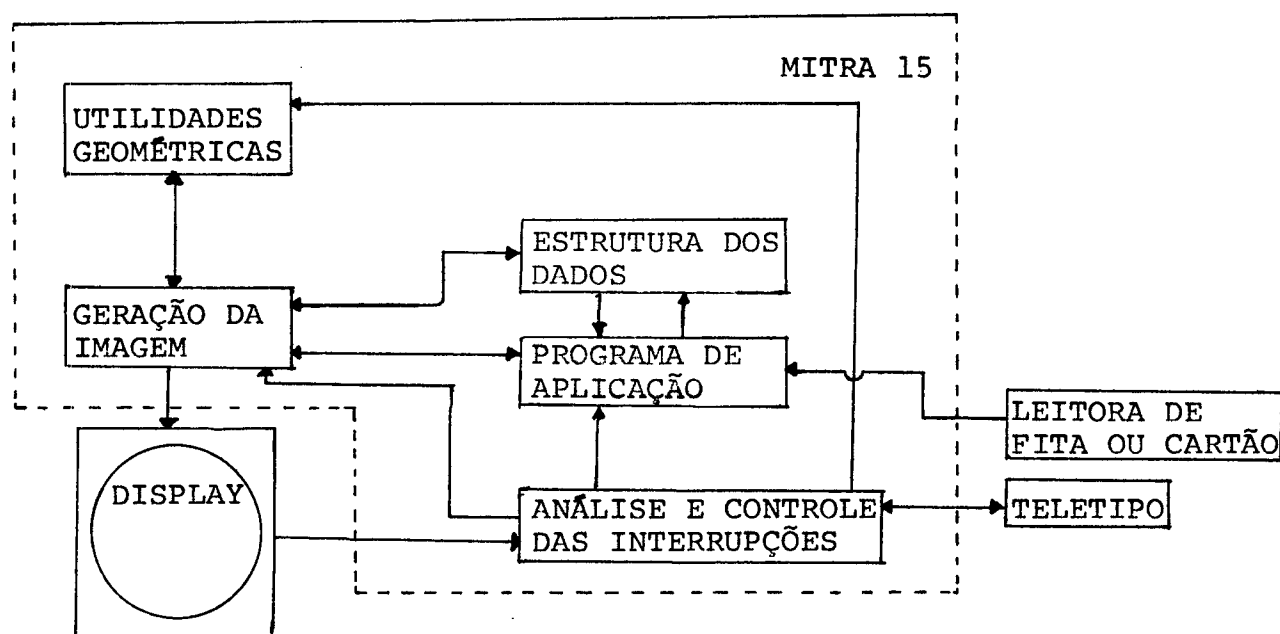


Figura 4.1

A figura 4.1 apresenta o esquema geral do sistema; o programa de aplicação se comunica com os blocos funcionais, conforme indicado no esquema, armazenando os dados do usuário e transmitindo os endereços destas subrotinas de utilidades geométricas ou geração de imagem, segundo se trate da manipulação de figuras tri ou bi-dimensionais.

Se as transformações geométricas forem requeridas, as rotinas de utilidades geométricas projetarão sobre um plano, as perspectivas das figuras tri-dimensionais fornecidas, eliminará as linhas escondidas e gerará um novo conjunto de dados que representará as figuras sob um plano.

As operações executadas pela geração de imagem, subdividem-se em duas categorias:

a) geração da lista do display: os dados representando figuras bi-dimensionais são processados por um conjunto de subrotinas que mapeiam o modelo do usuário, dentro das dimensões das coordenadas do display:

$$f(x,y) \rightarrow f'(x,y)$$

onde:

$$0 \leq x \leq 512 \quad \text{e} \quad 0 \leq y \leq 512 \quad \text{com } x,y \in \mathbb{N}$$

Palavras de controle são inseridas na lista do display com a finalidade de facilitar o manuseio desta pelo programa que controla a exibição da imagem sobre o vídeo.

b) Controle da imagem - é um programa que, operando no nível de interrupção do display, decodificará a lista do display e transmiti-

rã, via um conversor digital/analógico, os dados com os quais o display gerará as figuras sobre o tubo de raios catódicos - CRT .

A análise e controle das interrupções é função do sistema residente - MONITOR.

A indisponibilidade de um suporte, ao nível de hardware, que permita a instalação de dispositivos, tais como: light pen, key board, tracking cross, etc , acarretou a não iteratividade deste sistema. Qualquer modificação em um programa só será possível através de uma reprogramação ou modificação direta no programa em memória, antes que este seja lançado, utilizando - se o comando

```
% MODIFY / ....
```

que faz parte do acervo do sistema residente e funciona ao nível de interrupção do teletipo.

CAPÍTULO 5

As estruturas dos dados do sistema compreendem as formas canônicas (normalizadas, descrições de algoritmos matemático) dos objetos geométricos que deverão ser gerados, informações descritivas dos atributos, informações alfanuméricas e interrelações das diferentes porções (subdivisão) das imagens que serão exibidas.

Basicamente, podemos distinguir dois tipos de estruturas de dados:

- a) Estrutura do usuário
- b) Estrutura do display

5.1 - ESTRUTURA DO USUÁRIO

Os dados fornecidos ou gerados pelo programa de aplicação que irão compor a estrutura de dados do usuário se enquadram em uma destas duas categorias:

5.1.1 - NUMÉRICOS

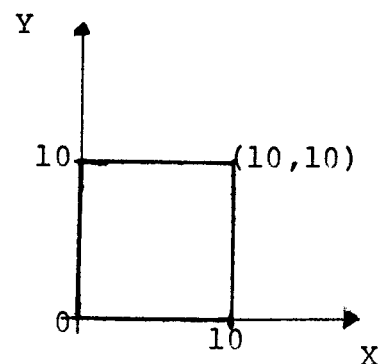
Constituídos por um conjunto de tabelas e subtabelas de $(3,N)$ elementos, sendo que cada tabela define uma subfigura e o conjunto de subtabelas define o corpo da figura principal.

Os elementos $(I=2,3 ; J=1,N)$ definem as coordenadas X e Y, respectivamente, dos extremos dos segmentos que formam uma figura. O elemento $(1,J)$ indica se o segmento a ser traçado é lu

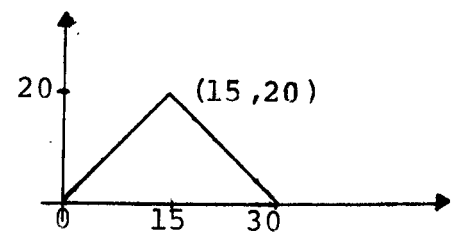
minoso, $((1,J)=1)$, ou não $((1,J)=0)$. N indica o número de segmentos que serão traçados.

As coordenadas de uma subfigura são expressas em relação a um referencial, cuja origem corresponde a um dos vértices ou ao centro da subfigura; a escolha da origem fica ao cargo do usuário e deve ser escolhida de modo a minimizar o número de segmentos a serem traçados, sem contudo prejudicar as características da subfigura. Exemplo:

	I →		
TAB1		1	10
	J	1	10
	↓	1	0
		1	0



TAB2		1	30
		1	15
		1	0



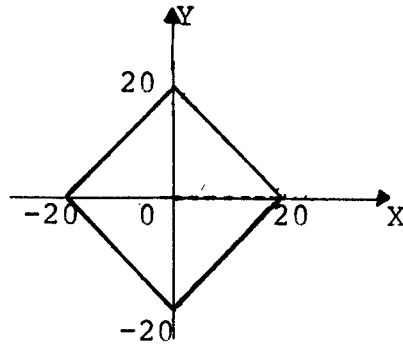
Os elementos $(I=1,2,3;1)$ determinam a condição e as coordenadas extremas do segmento cuja origem são as coordenadas $(0,0)$ do sistema tomado como referencial. Os elementos $(I=1,2,3;J)$ $(J=2,\dots,N)$, determinam a condição e as coordenadas extremas do segmento J , cuja origem são as coordenadas extremas do segmento $J-1$.

O extremo do último segmento que define uma subfigu

ra deve corresponder ao início do seu primeiro segmento. Caso esta condição não seja satisfeita, o sistema gera um segmento não luminoso que satisfará esta condição. Exemplo:

TAB3

0	20	0
1	0	20
1	-20	0
1	0	-20
1	20	0



Neste caso, o segmento gerado pelo sistema deslocará o raio luminoso da posição (20,0) para a posição (0,0) do sistema tomado como referencial.

Na geração de um círculo, o usuário apenas determinará o raio deste, então a subfigura que o definirá será gerada pelo sistema; a localização do círculo sobre o vídeo será determinada pelo usuário durante a definição do corpo da figura principal.

As coordenadas relacionadas à figura principal são expressas em relação ao sistema de coordenadas do display, isto é, seus valores variam no intervalo de (0 a 512).

As subtabelas definem um ou vários segmentos luminosos ou não, pertencentes à figura principal. O agrupamento das coordenadas da figura em subtabelas só é necessário quando as subfiguras são utilizadas. O número de subtabelas dependerá da complexidade da figura e do número de vezes que as subfiguras são conectadas à figura principal. Em princípio, a cada extremo do último

dos segmentos definido por uma subtabela será conectada uma subfigura, salvo se esta subtabela for a última do conjunto de subtabelas. Exemplos:

STAB1	0	50	40
	1	350	40
	1	350	400
	1	50	400

STAB2	1	350	400
-------	---	-----	-----

STAB3	0	350	40
-------	---	-----	----

STAB4	1	50	400
-------	---	----	-----

STAB5	0	200	400
	1	200	450

Para a primeira subtabela, os elementos $(I=1,2,3;1)$ definem a condição e as coordenadas extremas do segmento cuja origem são as coordenadas $(0,0)$ do display. Para as demais subtabelas os elementos $(I=1,2,3;1)$ definem a condição e as coordenadas extremas do último segmento cuja origem são as coordenadas extremas do último segmento da subtabela anterior. Para qualquer subtabela, os elementos $(I=1,2,3;J)$, $(J=2,3,\dots:N)$ definem a condição e as coordenadas extremas do segmento cuja origem são as coordenadas extremas do segmento $J-1$ pertencente à sequência de segmentos gerada pela subtabela.

O endereço de cada tabela ou subtabela corresponderá ao endereço de seu primeiro elemento.

5.1.2 - ALFANUMÉRICOS

Constituídos por cadeias de caracteres alfanuméricos destinados à geração de mensagens sobre o vídeo do display. Exemplo:

MENS "FIGURA GERADA PELO PROGRAMA DE APLICAÇÃO"

MENS é o label associado a um vetor de M posições que será preenchido com a mensagem, lida no formato A1, pelo programa do usuário.

5.2 - ESTRUTURA DO DISPLAY

O resultado do processamento dos dados fornecidos pelo usuário irão constituir a estrutura dos dados do display- lista do display.

O endereço (IMAG) do bloco de memória reservado para a armazenagem da lista do display é o endereço de sua primeira palavra. Um contador (RI), inicialmente igual a zero, apontará, relativo ao início do bloco, a primeira palavra vazia deste bloco, e seu acesso será feito através do endereçamento indireto geral indexado.

Inicialmente serão processadas todas as subfiguras, cujos endereços absolutos na lista do display serão armazenados na tabela de endereços previamente reservada pelo usuário.

O endereço absoluto da figura principal da lista do

display será armazenado na primeira palavra da zona comum do sistema (ZC), a fim de facilitar o acesso do programa de controle a esta estrutura.

Distinguiremos 7 tipos de informações básicas, constituindo a lista do display. Estas podem ser agrupadas em 2 categorias:

- a) Informações de controle;
- b) Informações a serem transmitidas ao display.

5.2.1 - INFORMAÇÕES DE CONTROLE

São constituídas por palavras duplas que determinam a ordem de exibição das informações que serão visualizadas sobre o vídeo. A primeira palavra indica a natureza do apontador; a segunda palavra aponta um endereço na lista do display.

al) Apontador Tipo 1

Sua função é indicar que o processamento de um bloco da lista do display (figura ou subfigura) foi completado e que sua segunda palavra constitui um apontador ao início deste bloco. Seu formato é o seguinte:

0	1	2	3
1	—	ENDEREÇO	

BYTE 0 - Indica a natureza do apontador

BYTE 1 - Não importa

BYTES 3,4 - Endereço de retorno ao início do bloco.

a2) Apontador Tipo 2

Indica a ruptura do processamento sequencial do bloco que define a figura principal e o endereço da subfigura para a qual o processamento é desviado. Neste interim, o endereço subsequente na figura principal é armazenado na segunda palavra do bloco que define a subfigura. Seu formato é o seguinte:

0	1	2	3
2	—	ENDEREÇO	

BYTE 0 - Indica a natureza do apontador

BYTE 1 - Não importa

BYTES 3,4 - Endereço da subfigura a exibir.

a3) Apontador Tipo 3

É constituído pelas duas primeiras palavras do bloco que define uma subfigura. Sua função é indicar o endereço de retorno à figura principal.

Na construção do bloco que define a subfigura, a segunda palavra deste apontador é deixada vazia; nela é armazenado o endereço de retorno à figura principal no momento em que o processamento da lista do display é deslocado para a subfigura. Seu formato é o seguinte:

5	—	VAZIA
---	---	-------

BYTE 0 - Indica a natureza do apontador

BYTE 1 - Não importa

BYTES 3,4 - Reservado ao endereço de retorno à figura principal.

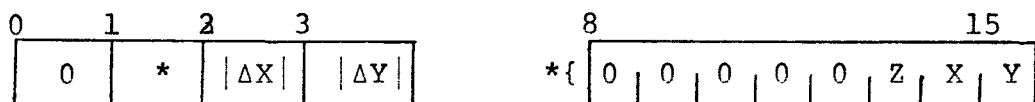
5.2.2 - INFORMAÇÕES A SEREM TRANSMITIDAS AO DISPLAY

São constituídas por palavras duplas ou uma série de N+1 palavras sequenciais, tais que a primeira palavra da série determina um número N de palavras que serão transmitidas.

Segundo as suas características, estas informações classificam-se em 4 tipos, sendo que 3 são resultados dos processamentos numéricos e o quarto resulta dos processamentos alfanuméricos.

b1) Informação Tipo 1

Determina as condições e o deslocamento do raio luminoso sobre o vídeo. A nova posição do raio luminoso é função da posição anterior e da informação recebida pelo display a qual ocupa uma palavra dupla, cujo formato é o seguinte:



BYTE 0 - Identifica a natureza da informação

BYTE 1 - Determina as condições do deslocamento

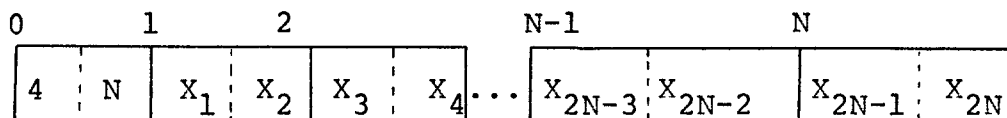
BIT 13 - Luminosidade $\begin{cases} Z = 1 & - \text{ Sim} \\ Z = 0 & - \text{ Não} \end{cases}$

BIT 14 - Sinal de ΔX $\begin{cases} X=1 & - \text{ Negativo} \\ X=0 & - \text{ Positivo} \end{cases}$

BIT 15 - Sinal de ΔY $\begin{cases} Y=1 & - \text{ Negativo} \\ Y=0 & - \text{ Positivo} \end{cases}$

b3) Informação Tipo 3

É constituída por uma sequência de $n+1$ palavras que definem uma mensagem a ser exibida sobre o vídeo. Os códigos dos caracteres alfanuméricos são armazenados 2 a 2, por palavra. Estes códigos são transmitidos ao display que, ao nível de hardware, gera os caracteres associados aos códigos recebidos. Seu formato é o seguinte:



PALAVRA 0:

BYTE 0 - Identifica a natureza da informação

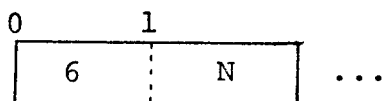
BYTE 1 - Indica o número de palavras ocupadas pela mensagem

PALAVRA 1 a N:

Contêm os $2N$ caracteres que formam a mensagem.

b4) Informação Tipo 4

Sua estrutura é análoga a da informação tipo 2; ela se destina à geração de círculos de raios iguais ou menores do que 12. Neste caso, as $n+1$ palavras sequenciais definem $1/4$ dos segmentos que gerarão o círculo. O formato da primeira palavra desta informação é o seguinte:



onde:

BYTE 0 - Identifica a natureza da informação

BYTE 1 - Indica o número de bytes seguintes que armazenam os segmentos.

5.3 - ORGANIZAÇÃO DA ESTRUTURA DO DISPLAY

A lista do display é organizada em blocos que definem as subfiguras e a figura principal. Os blocos das subfiguras apresentam em seu extremo superior um apontador do tipo 3 e no seu extremo inferior um apontador do tipo 1. As informações contidas entre estes apontadores podem ser dos tipos 1, 2, 3 e 4. Se a subfigura define uma mensagem, os incrementos $\Delta X=0$ e $\Delta Y=0$ serão transmitidos ao display. Isto acarretará o posicionamento do raio luminoso nas coordenadas (0,0) do display; em seguida o raio luminoso será deslocado para a posição determinada pelo usuário e então passa à geração da mensagem. Se a subfigura define um círculo, os incrementos $\Delta X=\text{Raio}$ e $\Delta Y=0$ deslocará o raio luminoso do centro do círculo até a extremidade do seu raio, e então passa a gerar os N segmentos.

O bloco da figura principal apresenta no seu extremo superior uma informação do tipo 1, cujos incrementos $\Delta X=0$ e $\Delta Y=0$ têm a função de posicionar o raio luminoso nas coordenadas (0,0) do display e eliminar os erros acumulados no display durante a geração das sequências de segmentos luminosos. No seu extremo inferior, este bloco apresenta um apontador do tipo 1 que assegura a restauração da imagem. As informações contidas neste bloco são do tipo

1 e apontadores do tipo 2.

A figura 5.1 apresenta uma visão geral da estrutura da lista do display; o início de cada bloco que define uma subfigura ou a figura principal é destacado pelo endereço simbólico escrito ao lado.

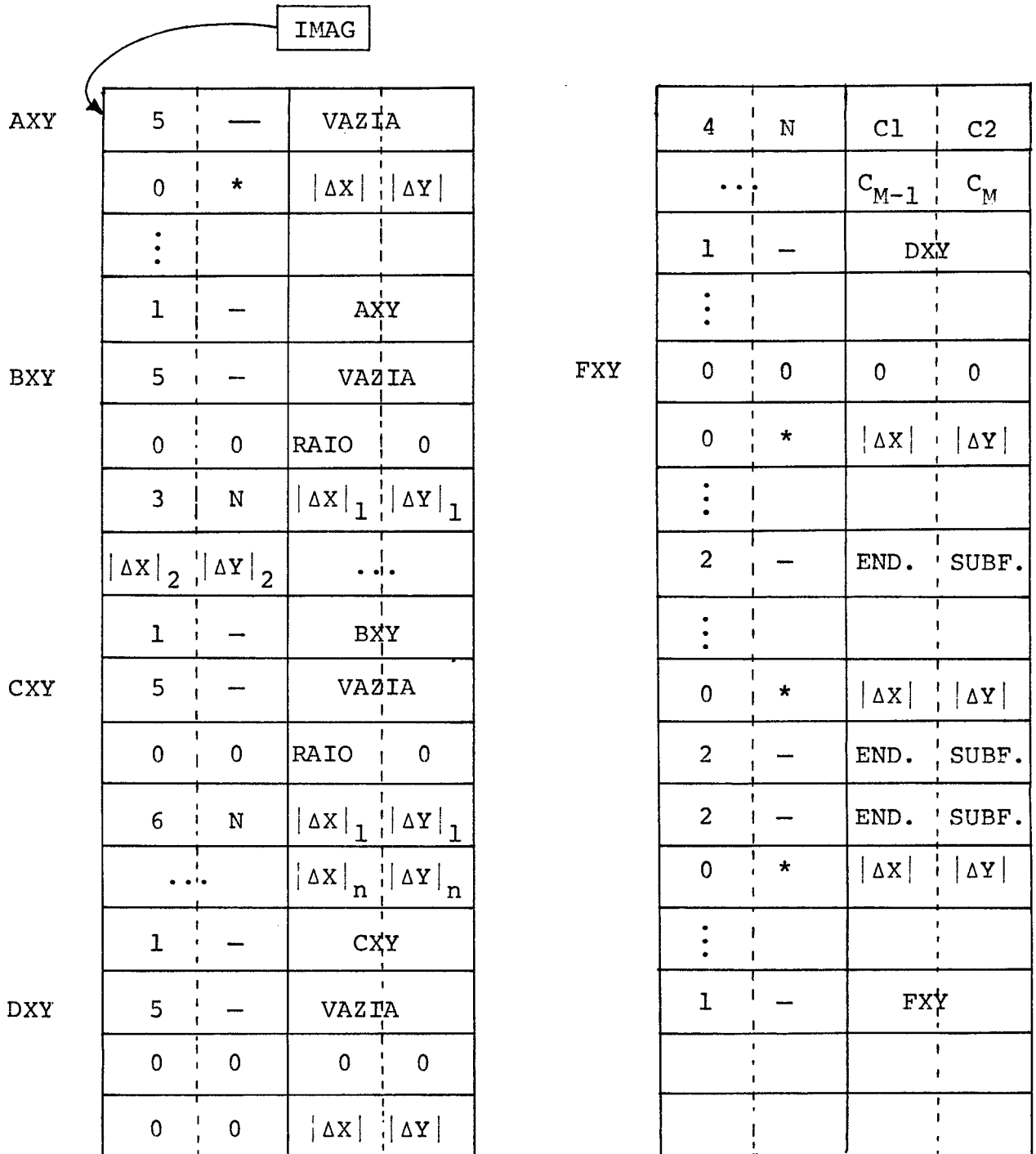


FIGURA 5.1

CAPÍTULO 6

GERAÇÃO DE FIGURAS

Sabemos que a geração das figuras subdivide-se em duas partes distintas: geração da lista do display e o controle do processamento desta lista.

Inicialmente, trataremos da exposição detalhada do conjunto de subrotinas destinadas à construção da lista do display e posteriormente detalharemos o programa de controle e exibição da imagem.

6.1 - SUBROTINAS BÁSICAS

A fim de facilitar a compreensão dos algoritmos e a programação destes, transcreveremos o significado dos símbolos que serão utilizados:

ACC - Acumulador

EXT - Extensão do acumulador

RX - Registro de índice

X - Conteúdo do RX

END - Endereço

@END - Endereço indireto

@END,X - Endereço indireto indexado

IMAG - Ponteiro para a área de memória da lista do display

RI - Contador que indica a próxima palavra vazia na área do display

Cl - Abscissa de origem do segmento

C2 - Ordenada de origem do segmento

LD - Lista do Display

6.1.1 - DEFIG: Definição de figura ou subfigura

Sua função é gerar o apontador tipo 3 ou a informação tipo 1, com $\Delta X=0$ e $\Delta Y=0$, no início do bloco de uma subfigura / ou figura e armazenar no endereço fornecido pelo usuário, a localização deste bloco na lista do display.

A chamada desta subrotina obedecerá à seguinte nota -
ção:

CALL DEFIG (END,IND)

onde:

END - Palavra reservada à armazenagem do endereço da figura ou subfigura.

IND - Indica se será definida uma figura (IND=0) ou subfigura (IND=1).

Exemplo: CALL DEFIG (END1,0) define uma subfigura cuja localização na lista do display é armazenada em END1.

ALGORITMO DA SUBROTINA DEFIG

1 - INICIO

2 - DETERMINAR O ENDEREÇO DO BLOCO NA LD E ARMAZENÁ-LO NO EN
DEREÇO FORNECIDO

3 - SE FOR FIGURA, GERE A INFORMAÇÃO TIPO 1 C/ $\Delta X=0$ E $\Delta Y=0$,
VÁ PARA 5; SE NÃO, VÁ PARA 4

4 - GERE APONTADOR TIPO 3

- 5 - ATUALIZE O CONTADOR RI
- 6 - RETORNO
- 7 - FIM

6.1.2 - FIM: Finaliza uma figura ou subfigura

Sua função é gerar um apontador tipo 1 nas duas últimas palavras do bloco que define uma figura ou subfigura; antes e-la testa se as variáveis C1 e C2 estão zeradas, caso contrário, é gerada uma informação tipo 1 com $\Delta X = -C1$ e $\Delta Y = -C2$. Então são zeradas as variáveis C1 e C2 e em seguida é construído o apontador.

A chamada desta subrotina obedece à seguinte notação:

CALL FIM (END,IND)

onde:

END - Endereço da figura ou subfigura na lista do display

IND - Indica a finalização de uma figura (IND=1) ou subfigura (IND=0)

Exemplo: CALL (ENDF,1) finaliza a figura principal da lista do display cujo endereço está armazenado em ENDF.

ALGORITMO DA SUBROTINA FIM

- 1 - INICIO
- 2 - SE FOR FIGURA VÁ PARA 7 ; SE NÃO, VÁ PARA 3
- 3 - SE C1=0 VÁ PARA 4 ; SE NÃO VÁ PARA 5
- 4 - SE C2=0 VÁ PARA 8 ; SE NÃO VÁ PARA 5
- 5 - CARREGAR (-C2) NO REGISTRO EXTENSÃO E CARREGAR (-C1) NO REGISTRO ACC

```

6 - CHAMAR  CINT1
7 - ZERAR  C1  E  C2
8 - GERAR APONTADOR TIPO 1
9 - ATUALIZAR  RI
10 - RETORNO
11 - FIM

```

6.1.3 - LIGSF: Conecta a subfigura à figura principal da lista do Display

Sua função é gerar um apontador tipo 2 na figura principal da lista do display; conforme já vimos, este apontador desvia o processamento para a subfigura cujo endereço se encontra na sua segunda palavra.

A chamada desta subrotina obedece à seguinte notação:

```
CALL LIGSF (END)
```

onde:

END - Endereço da subfigura na lista do display

ALGORITMO DA SUBROTINA LIGSF

```

1 - INICIO
2 - GERAÇÃO DA PRIMEIRA PALAVRA DO APONTADOR TIPO 2
3 - ARMAZENAR O ENDEREÇO DA SUBFIGURA NA SEGUNDA PALAVRA DO
APONTADOR
4 - ATUALIZAR  RI
5 - RETORNO
6 - FIM

```

6.1.4 - GEFIG : Gera as figuras ou subfiguras

A função desta subrotina é processar a tabela fornecida pelo usuário, resultando a sequência de segmentos que definem uma subfigura ou parte da figura principal. Esta subrotina chama a subrotina CINT1 , cuja função será mais tarde abordada.

Devemos ressaltar que na programação desta e das demais subrotinas, aproveitamos o fato do sistema residente - MONITOR - inicializar todas as variáveis e áreas que são reservadas, fazendo-as iguais a zero; em vista disto, quando tivermos que armazenar zeros na lista do display, simplesmente incrementaremos o contador RI .

A chamada desta subrotina obedece à seguinte notação:

```
CALL GEFIG (END,NUM)
```

onde:

END - Endereço da tabela ou subtabela

NUM - Número de segmentos que serão gerados

Exemplo:

```
CALL GEFIG (TAB1,5)
```

TAB1 é o endereço da tabela de (3,5) elementos que define uma subfigura constituída de 5 segmentos.

ALGORITMO DA SUBROTINA GEFIG

1 - INICIO

2 - POSICIONAR O BIT QUE INDICA SE O SEGMENTO É LUMINOSO OU NÃO

```

3 - CALCULAR  $\Delta Y$ 
4 - ATUALIZAR C1
5 - CALCULAR  $\Delta Y$ 
6 - ATUALIZAR C2
7 - CHAMAR CINT1
8 - SE FOR O ÚLTIMO SEGMENTO, VÁ PARA 9 ;
SE NÃO, VÁ PARA 2
9 - RETORNO
10 -FIM

```

6.1.5 - CINT1 : Complemento Informação Tipo 1

Este subprograma será chamado pela subrotina GEFIG e ocasionalmente pela subrotina FIM, dependendo das condições de finalização de uma subfigura, conforme anteriormente já foi explicado.

Ela posicionará os bits que indicam os sinais de ΔX e ΔY e armazenará os valores absolutos de ΔX e ΔY na segunda palavra da informação tipo 1 que representará o segmento em apreço.

Caso os valores absolutos de ΔX e ΔY ultrapassem 255 (valor máximo permitido pelo octeto ocupado por cada incremento, serão geradas duas informações tipo 1, iguais, tais que os incrementos serão $|\Delta X|:2$ e $|\Delta Y|:2$. Neste caso, os incrementos ΔX e ΔY da segunda informação tipo 1 serão acrescidos do conteúdo da posição de memória PXY, de forma que o primeiro byte (PX) desta palavra conterà o valor um(1) quando ΔX for ímpar e o valor zero (0) quando ΔX for par. O segundo byte (PY) apresentará os mesmos valo

res para análogas condições de ΔY . Isto eliminará possíveis erros de arredondamentos resultantes da divisão por 2 executadas sobre os valores de ΔX e ΔY que obrigatoriamente ocorrerão quando estes valores forem ímpares.

Os valores de ΔX e ΔY serão transmitidos a este subprograma via os registros acumulador e extensão do acumulador.

ALGORITMO DA SUBROTINA CINT1

```

1 - INICIO
2 - POSICIONAR OS BITS DO SINAL DE  $\Delta X$  E  $\Delta Y$ 
3 - SE  $\Delta X < 255$  VÁ PARA 4; SE NÃO VÁ PARA 7
4 - SE  $\Delta Y < 255$  VA PARA 5 ; SE NÃO VÁ PARA 7
5 - REUNIR  $\Delta X$  E  $\Delta Y$  NO ACC
6 - ARMAZENAR  $\Delta X$  E  $\Delta Y$  NA LD E VÁ PARA 13
7 - SE  $\Delta Y$  É ÍMPAR FAÇA  $PY=1$  E VÁ PARA 8; SE NÃO , VÁ
PARA 8
8 - SE  $\Delta X$  É IMPAR, FAÇA  $PX=1$  E VÁ PARA 9; SE NÃO , VÁ
PARA 9
9 - FAÇA  $\Delta X = \Delta X:2$  E  $\Delta Y = \Delta Y:2$ 
10 - REUNIR  $\Delta X$  E  $\Delta Y$  NO ACC
11 - ARMAZENAR  $\Delta X$  E  $\Delta Y$  NA LD
12 - DUPLICAR A INFORMAÇÃO TIPO 1 GERADA, SOMANDO-LHE O CON-
TEÚDO DE  $PXY$ 
13 - ATUALIZAR RI
14 - RETORNO
15 - FIM

```

6.1.6 - CIRCO : Definição de Círculo

Esta subrotina foi programada em FORTRAN. Se o raio do círculo foi maior que 12, ela chama a subrotina DEFIG e em seguida gera uma sequência de pares $(|\Delta X|, |\Delta Y|)$, tais que, após cada par gerado, se este não for igual a $(|\Delta X|=0, |\Delta Y|=0)$, é chamada a subrotina CIRC. O número de pares gerados corresponderá a um oitavo dos segmentos que definirão o círculo; após a geração do último segmento será chamada a subrotina FIM. Se o raio foi menor ou igual a 12, apenas será chamada a subrotina CIRC. Neste caso, as subfiguras que definem círculos de raio menor ou igual a 12, se encontram tabeladas e apenas seu endereço é armazenado no endereço fornecido pelo usuário.

O número de pares $(|\Delta X|, |\Delta Y|)$ que serão calculados é determinado pela expressão:

$$NSEG = RAOIO : 4 + 1$$

As coordenadas que definirão os segmentos serão expressas em relação a um sistema de coordenadas cuja origem coincide com o centro do círculo. Assim, as coordenadas de origem do primeiro segmento será (Raio, 0). Para os demais segmentos, suas origens serão as coordenadas extremas do segmento anterior. As coordenadas extremas dos segmentos serão determinadas usando-se a propriedade trigonométrica dos triângulos retângulos:

$$X = \text{Raio} \times \cos(\beta)$$

e a equação do círculo com centro na origem:

$$Y = \sqrt{(RAIO)^2 - X^2}$$

Para o primeiro segmento, o valor de β será:

$$\beta = \frac{\pi}{4} \times \frac{1}{NSEG}$$

Para os demais segmentos, o valor de β será:

$$\beta = \beta + \alpha$$

sendo α igual ao valor inicial de β .

A chamada desta subrotina obedece à seguinte notação:

CALL CIRCO (END, RAI0)

onde:

END - Endereço onde será armazenado o endereço do círculo na lista do display

RAIO - Valor do raio do círculo

Exemplo:

CALL CIRCO (END2,30)

ALGORITMO DA SUBROTINA CIRCO

- 1 - INICIO
- 2 - CHAMAR DEFIG
- 3 - SE RAI0 < 12 CHAMAR CRIMD E VÁ PARA 24,
SE NÃO VÁ PARA 4 .
- 4 - CALCULAR O NÚMERO DE SEGMENTOS: NSEG
- 5 - INICIALIZAR AS COORDENADAS DE ORIGEM
- 6 - CALCULAR O ÂNGULO INICIAL
- 7 - CÁLCULO DAS COORDENADAS EXTREMAS
- 8 - CALCULAR $|\Delta X|$ e $|\Delta Y|$

```

9 - SE  $|\Delta X|=0$  VÁ PARA 10 ; SE NÃO VÁ PARA 11
10- SE  $|\Delta Y|=0$  VÁ PARA 14 ; SE NÃO VÁ PARA 11
11- ATUALIZAR AS COORDENADAS DE ORIGEM
12- ARMAZENAR  $\Delta X$  E  $\Delta Y$  NOS PARÂMETROS DE CIRC
13- CHAMAR CIRC
14- FAÇA NSEG=NSEG - 1
15- SE NSEG = 1 VÁ PARA 18 ; SE NÃO VÁ PARA 16
16- INCREMENTAÇÃO DO ÂNGULO
17- FAÇA CONTROLE DE CIRC= 0 E VÁ PARA 7
18- FAÇA COORDENADAS EXTREMAS = RAO /  $\sqrt{2}$ 
19- CALCULAR  $|\Delta X|$  E  $|\Delta Y|$ 
20- SE  $|\Delta X|=0$  VÁ PARA 21 ; SE NÃO VÁ PARA 22
21- SE  $|\Delta Y|=0$  VÁ PARA 24; SE NÃO VÁ PARA 22
22- ARMAZENAR  $|\Delta X|$  E  $|\Delta Y|$  NOS PARÂMETROS DE CIRC
23- CHAMAR CIRC
24- CHAMAR FIM
25- RETORNO
26- FIM

```

6.1.7 - CIRC : Geração da Informação Tipo 3

A primeira vez que esta subrotina é chamada, ela constrói a primeira palavra da informação tipo 3 que será gerada e armazena na palavra seguinte os valores de $|\Delta X|$ e $|\Delta Y|$. Nas demais chamadas, ela armazena os valores de $|\Delta X|$ e $|\Delta Y|$ e incrementa o contador localizado no segundo byte da primeira palavra da informação tipo 3. Este contador indicará o número de pares ($|\Delta X|$,

$|\Delta Y|$) que foram armazenados na lista do display, cada par ocupando uma palavra.

A chamada desta subrotina obedece à seguinte notação:

CALL CIRC (IX,IY,RAIO, IND)

onde:

IX - Valor absoluto de ΔX

IY - Valor absoluto de ΔY

RAIO - Valor do raio do círculo

IND - Parâmetro de controle. Na primeira chamada, IND=1; nas demais, IND=0 .

ALGORITMO DA SUBROTINA CIRC

1 - INICIO

2 - SE CONTROLE DE CIRC=1 , VÁ PARA 3 ; SE NÃO, VÁ PARA 4

3 - CONSTRUIR PRIMEIRA PALAVRA INFORMAÇÃO TIPO 3 , DEIXANDO SEU SEGUNDO BYTE ZERADO.

4 - REUNIR $|\Delta X|$ E $|\Delta Y|$ NO ACC

5 - ARMAZENAR $|\Delta X|$ $|\Delta Y|$ NA LD

6 - SOMAR (+2) AO CONTEÚDO DO SEGUNDO BYTE DA PRIMEIRA PALAVRA DA INFORMAÇÃO TIPO 3

7 - ATUALIZAR RI

8 - RETORNO

9 - FIM

6.1.8 - Subrotina CRIMD : Círculos de Raio ≤ 12

Esta subrotina manipula as estruturas compactadas

(conforme mostradas na listagem - Vide Apêndice) dos círculos de raio ≤ 12 e gera na lista do display - LD - uma informação tipo 4.

Verificou-se que o algoritmo desenvolvido para calcular os pares $(|\Delta X|, |\Delta Y|)$, (Subrotina CIRCO), tornava-se impreciso para valores de raio ≤ 12 , o que acarretaria grosseiras deformações na geração destes círculos. Em vista disto, resolveu-se guardar, compactados, os segmentos que os definirão. Em uma palavra desta estrutura compactada, pode-se armazenar até 4 pares $(|\Delta X|, |\Delta Y|)$, utilizando-se um campo de 2 bits para cada incremento ΔX ou ΔY .

Na determinação dos segmentos, verificou-se também de formações na geração dos círculos se fôsse tomada uma sequência de segmentos correspondente a um oitavo do círculo. Resolveu-se este problema criando-se novo tipo de informação (Informação Tipo 4), cuja sequência de segmentos definiriam um quadrante do círculo.

A chamada desta subrotina obedece à seguinte notação:

CALL CRIMD (END)

onde:

END - Endereço do raio do círculo

ALGORITMO DA SUBROTINA CRIMD

1 - INICIO

2 - ARMAZENAR NA LD O SEGMENTO QUE DESLOCA O RAIOS LUMINOSOS PARA A PERIFERIA DO CÍRCULO

3 - GUARDAR O ENDEREÇO DA PRIMEIRA PALAVRA TIPO 4

4 - COLETAR O ENDEREÇO DA TABELA COMPACTADA DO RAIOS FORNECIDA

DO

```

5 - FAZER COM=0
6 - COLETAR UMA PALAVRA DE ESTRUTURA COMPACTADA
7 - DECODIFICAR UM PAR ( $|\Delta X|$ ,  $|\Delta Y|$ )
8 - SE  $|\Delta X|$  E  $|\Delta Y|$  FOREM IGUAIS A ZERO, VÁ PARA 12; SE NÃO,
VÁ PARA 9
9 - ARMAZENAR O PAR ( $|\Delta X|$ ,  $|\Delta Y|$ ) NA LD
10- FAÇA CON = CON + 1
11- SE COM=4 VÁ PARA 5 ; SE NÃO, VÁ PARA 7
12- CALCULAR O NÚMERO DE PARES ( $|\Delta X|$ ,  $|\Delta Y|$ ) ARMAZENADO NA LIS
TA DO DISPLAY
13- CONSTRUIR PRIMEIRA PALAVRA INFORMAÇÃO TIPO 4
14- ATUALIZAR RI
15- RETORNO
16 - FIM

```

6.1.9 - GEMEN: Geração de Mensagem

Uma mensagem será definida por uma subfigura construída pelas chamadas em sequência, das subrotinas DEFIG, GEMEN e FIM. Neste tipo de subfigura, a primeira informação tipo 1 que segue ao apontador tipo 3 terá os incrementos $\Delta X=0$ e $\Delta Y=0$. A informação seguinte, da mesma natureza, posicionará o raio luminoso nas coordenadas do display fornecidas pelo usuário.

A função desta subrotina é gerar na lista do display, as duas informações tipo 1 supra citadas e a informação tipo 3 que define o corpo da mensagem.

Caso uma mensagem seja constituída por um número im -

par de caracteres, então o caracter branco será incluído no fim da mensagem.

Uma informação tipo 3 no máximo poderá ocupar 256 palavras, isto é, uma mensagem de até 510 caracteres pode ser gerada.

A chamada desta subrotina obedece à seguinte notação:

CALL GEMEN (CAD,X,Y,NUM)

onde:

CAD - Endereço da cadeia de caracteres

X - Abscissa do início da mensagem sobre o vídeo

Y - Ordenada do início da mensagem sobre o vídeo

NUM - Número de palavras, cada uma com 2 caracteres, ocupadas na lista do display

ALGORITMO DA SUBROTINA GEMEN

1 - INICIO

2 - CONSTRUIR PRIMEIRA INFORMAÇÃO TIPO 1

3 - SE X > 255, VÁ PARA 5 ; SE NÃO, VÁ PARA 4

4 - SE Y > 255, VÁ PARA 5 ; SE NÃO, VÁ PARA 6

5 - CHAMAR CINT1; VÁ PARA 7

6 - CONSTRUIR SEGUNDA INFORMAÇÃO TIPO 1

7 - ARMAZENAR DOIS CARACTERES NA LD

8 - FAÇA NUM = NUM - 1

9 - SE NUM > 0 VÁ PARA 7 ; SE NÃO, VÁ PARA 10

10- ATUALIZAR RI

11- RETORNO

12 -FIM

6.1.10 - LPCEI : Lançamento do Programa de Controle e Exibição da Imagem

A função desta subrotina é armazenar na primeira posição da ZONA COMUM (ZC) o endereço absoluto da figura principal da lista do display e chamar o módulo do sistema residente responsável pela demanda das operações sobre o sistema de interrupções (M:IT), que lançará o programa de controle e exibição da imagem conectado ao nível de interrupção do display.

Esta subrotina não devolverá o processamento ao programa principal, visto que suas últimas instruções constituem o laço de espera das sequentes ativações do nível de interrupção do display que serão geradas automaticamente pelo próprio display.

Na LDS desta subrotina será definido o contador RI e reservada a área IMAGE destinada à construção da lista do display.

Sua chamada deve ser o último comando do programa do usuário e obedece à seguinte notação:

CALL LPCEI (END)

onde:

END - Endereço da figura principal na lista do display

ALGORITMO DA SUBROTINA LPCEI

- 1 - INICIO
- 2 - ARMAZENAR O ENDEREÇO DA FIGURA PRINCIPAL EM ZC
- 3 - CHAMAR M:IT
- 4 - LAÇO DE ESPERA DA INTERRUPÇÃO DO NÍVEL DO DISPLAY
- 5 - FIM

6.2 - PROGRAMA DE CONTROLE E EXIBIÇÃO DA IMAGEM

Sabemos que a lista do display é constituída por diversos tipos de informações e apontadores e que todos os endereços constantes nos seus apontadores são absolutos; além disso, o tempo gasto na transferência de uma informação da lista do display aos seus registros é bastante inferior ao que o display gasta para liberar estes registros a novas informações.

A chave contida no primeiro byte das informações ou apontadores determina que ações devem ser executadas. O funcionamento deste programa e a interpretação da lista do display consiste basicamente em testar estas chaves.

O primeiro lançamento deste programa é determinado pela execução da subrotina LPCEI; neste ínterim, a informação tipo 1 que inicia o bloco da figura principal é mandada diretamente aos registros do display; a informação seguinte será armazenada nos registros R1 e R2 e então o programa será desativado. Os seguintes lançamentos serão controlados pelo próprio display que ativará o programa sempre que os seus registros estejam habilitados a receberem nova informação. O programa assim ativado, transferirá o conteúdo dos registros R1 e R2 aos registros do display, armazenará nova informação nos registros R1 e R2 e se desativará.

Serão usadas como registros do display a segunda e a terceira palavra da zona comum do sistema. O acesso a estas posições de memória far-se-á através do endereçamento indireto geral indexado, utilizando-se a instrução de suplo armazenamento (DST-Dou-

ble Store) e o valor 2 carregado no registro de índice (RX).

No processamento das informações tipo 3 e tipo 4, o registro RX controlará o laço das chamadas da subrotina EXEC. O registro extensão transferirá o endereço do primeiro par $(|\Delta X|, |\Delta Y|)$; o primeiro byte do acumulador indicará se haverá ou não uma inversão no par $(|\Delta X|, |\Delta Y|) : (|\Delta X| = |\Delta Y|; |\Delta X| = |\Delta Y|)$. O segundo byte do acumulador indicará se as n palavras sequenciais serão coletadas (quanto à sua localização na memória do computador) em ordem crescente ou decrescente. Evidentemente, no processamento da informação tipo 4, o primeiro byte do acumulador será sempre igual a zero.

No processamento da informação tipo 3, os dois caracteres guardados em cada palavra serão armazenados separadamente nos registros R1 e R2. O bit zero (o mais forte) destas posições de memória será posicionado com o valor 1 indicando que elas contêm caracteres. Em seguida, o nível do display é desativado. Na reativação deste nível, o conteúdo de R1 e R2 é transferido aos registros do display; os registros R1 e R2 são preenchidos com dois novos caracteres e novamente o nível do display é desativado. Repete-se o ciclo até que a informação tipo 3 é completamente processada.

O algoritmo em seguida apresentado é auto-elucidativo quanto aos demais detalhes do funcionamento deste programa.

ALGORITMO DO PROGRAMA DE CONTROLE E EXIBIÇÃO DA IMAGEM

1 - INICIO

- 2 - COLETAR O ENDEREÇO DA FIGURA PRINCIPAL NA ZC
- 3 - LER O VALOR DA BASE G
- 4 - CALCULAR O ENDEREÇO DA FIGURA PRINCIPAL EM RELAÇÃO À BASE G
- 5 - TRANSFERIR A INFORMAÇÃO TIPO 1 APONTADA POR GI AO REGISTRO DO DISPLAY
- 6 - INCREMENTAR O CONTEÚDO DO APONTADOR GI DE 4 UNIDADES
- 7 - SE O PRIMEIRO BYTE APONTADO POR GI FOR IGUAL A ZERO, VÁ PARA 13; SE NÃO VÁ PARA 8
- 8 - SE O PRIMEIRO BYTE APONTADO POR GI FOR IGUAL A UM, VÁ PARA 17; SE NÃO, VÁ PARA 9
- 9 - SE O PRIMEIRO BYTE APONTADO POR GI FOR IGUAL A DOIS, VÁ PARA 20; SE NÃO, VÁ PARA 10
- 10- SE O PRIMEIRO BYTE APONTADO POR GI FOR IGUAL A TRÊS, VÁ PARA 27; SE NÃO, VÁ PARA 11
- 11- SE O PRIMEIRO BYTE APONTADO POR GI FOR IGUAL A QUATRO, VÁ PARA 40; SE NÃO, VÁ PARA 12
- 12- SE O PRIMEIRO BYTE APONTADO POR GI FOR IGUAL A CINCO, VÁ PARA 48; SE NÃO, VÁ PARA 50
- A - PROCESSAMENTO DA INFORMAÇÃO TIPO 1
 - 13- TRANSFERIR A INFORMAÇÃO APONTADA POR GI AOS REGISTROS R1 E R2
 - 14- DESATIVAR O NÍVEL DO DISPLAY
 - 15- TRANSFERIR O CONTEÚDO DE R1 E R2 AOS REGISTROS DO DISPLAY
 - 16- VÁ PARA 6
- A1 - PROCESSAMENTO APONTADOR TIPO 1
 - 17- CALCULAR RELATIVO À BASE G O ENDEREÇO APONTADO POR GI+2

18 - ARMAZENAR O ENDEREÇO CALCULADO EM GI

19 - VÁ PARA 7

A2 - PROCESSAMENTO APONTADOR TIPO 2

20 - CARREGAR EM RX O VALOR DE GI E SOMAR (+4)

21 - CALCULAR RELATIVO À BASE G O ENDEREÇO APONTADO POR GI+2

22 - ARMAZENAR O ENDEREÇO CALCULADO EM GI

23 - INCREMENTAR O APONTADOR GI DE 2 UNIDADES

24 - ARMAZENAR RX NO ENDEREÇO APONTADO POR GI

25 - INCREMENTAR O APONTADOR GI DE 2 UNIDADES

26 - VÁ PARA 7

A3 - PROCESSAMENTO INFORMAÇÃO TIPO 2

27 - ARMAZENAR EM UMA PALAVRA RESERVADA N A MEMÓRIA (LONT)

O CONTEÚDO DO ENDEREÇO APONTADO POR GI+1

28 - FAÇA $GI=GI+2$; $RX=0$; $EXT=GI$

29 - FAÇA $ACC=0$

30 - CHAMAR EXEC

31 - FAÇA 1º BYTE $ACC=1$; 2º BYTE $ACC=1$

32 - CHAMAR EXEC

33 - FAÇA $RX=RX+1$; 2º BYTE $ACC=1$

34 - CHAMAR EXEC

35 - FAÇA 1º BYTE $ACC=1$

36 - SE $RX=3$, VÁ PARA 38 ; SE NÃO, VÁ PARA 37

37 - FAÇA $RX=2$ E VÁ PARA 29

38 - SOMAR A GI O CONTEÚDO DE CONT

39 - VÁ PARA 7

A4 - PROCESSAMENTO INFORMAÇÃO TIPO 3

40 - ARMAZENAR EM UMA PALAVRA RESERVADA NA MEMÓRIA (CONT) O CONTEÚDO DO ENDEREÇO APONTADO POR GI+1

41 - FAÇA $GI = GI+2$

42 - ARMAZENAR EM R1 O CARACTER APONTADO POR GI E FAZER O BIT ZERO (O MAIS FORTE) DE R1 IGUAL A UM

43 - FAÇA $GI = GI+1$

44 - ARMAZENAR EM R2 O CARACTER APONTADO POR GI E FAZER O BIT ZERO DE R2 IGUAL A UM

45 - DESATIVAR O NÍVEL DO DISPLAY

46 - FAÇA $GI = GI+1$; $CONT = CONT - 1$

47 - SE $CONT = 0$, VÁ PARA 7 ; SE NÃO, VÁ PARA 42

A5 - PROCESSAMENTO APONTADOR TIPO 3

48 - ARMAZENAR EM GI O ENDEREÇO APONTADO POR GI+2

49 - VÁ PARA 7

A6 - PROCESSAMENTO INFORMAÇÃO TIPO 4

50 - ARMAZENAR EM CONT O CONTEÚDO APONTADO POR GI+1

51 - FAÇA $GI = GI+2$; $RX = 0$; $EXT = GI$

52 - FAÇA $ACC=0$

53 - CHAMAR EXEC

54 - FAÇA $RX = RX+1$; $ACC = 1$

55 - CHAMAR EXEC

56 - FAÇA $RX = RX+1$

57 - SE $RX = 4$, VÁ PARA 38 ; SE NÃO, VÁ PARA 52

58 - FIM

6.2.1 - SUBROTINA EXEC : EXIBIÇÃO DO CÍRCULO

Os parâmetros desta subrotina lhe são transmitidos via os registros: acumulador, extensão do acumulador e registro de índice. Em vista disto, ela inicia salvando estes parâmetros em posições pré-estabelecidas conforme mostramos no seu algoritmo.

O primeiro byte do acumulador sendo igual a zero, indica que o par ($|\Delta X|, |\Delta Y|$) não sofrerá modificações; se este byte fôr igual a 1, os valores de $|\Delta X|$ e $|\Delta Y|$ serão comutados.

O segundo byte do acumulador sendo igual a zero, indica que a sequência das n palavras será percorrida em ordem crescente; se igual a 1, em ordem decrescente.

A palavra que completará a informação tipo 1 (especificando os sinais de $|\Delta X|$ e $|\Delta Y|$ e a luminosidade originada de cada par ($|\Delta X|, |\Delta Y|$) é coletada de uma tabela. O conteúdo do registro de índice além de controlar o laço do programa principal que chama esta subrotina, também determina o acesso à tabela acima citada.

O número N de pares ($|\Delta X|, |\Delta Y|$) que constitui um oitavo ou um quarto dos segmentos que definem o círculo é armazenado na posição de memória CONT1. Este valor é restaurado cada vez que EXEC é chamado.

ALGORITMO DA SUBROTINA EXEC

1 - INICIO

2 - ARMAZENAR EM UM BYTE RESERVADO NA MEMÓRIA (A1) O CONTEÚDO DO 1º BYTE DO ACC

3 - ARMAZENAR EM UM BYTE RESERVADO NA MEMÓRIA (A2) O CONTEÚDO DO 2º BYTE DO ACC

4 - ARMAZENAR EM UMA PALAVRA RESERVADA NA MEMÓRIA (DISP) O CONTEÚDO DE EXT.

5 - SE $A1 = 0$, VÁ PARA 7 : SE NÃO, VÁ PARA 6

6 - COMUTAR OS VALORES DE ΔX E ΔY

7 - ARMAZENAR O PAR ($|\Delta X|$, $|\Delta Y|$) EM R2

8 - COLETAR NA TABELA DE CONDIÇÕES A PALAVRA QUE COMPLETARÁ A INFORMAÇÃO TIPO 1 E ARMAZENÁ-LA EM R1

9 - DESATIVAR O NÍVEL DO DISPLAY

10- TRANSFERIR O CONTEÚDO DE R1 E R2 AOS REGISTROS DO DISPLAY

11- SE FOI PROCESSADO O ÚLTIMO PAR ($|\Delta X|$, $|\Delta Y|$), VÁ PARA 15 ;
SE NÃO, VÁ PARA 12

12 - SE $A2=0$, VÁ PARA 13 ; SE NÃO VÁ PARA 14

13- FAÇA $DISP = DISP + 2$ E VÁ PARA 5

14 - FAÇA $DISP = DISP - 2$ E VÁ PARA 5

15 - RETORNO

16 - FIM

CONCLUSÃO

Conforme anteriormente citamos, o sistema aqui descrito, em vista da carência de um suporte de hardware apropriado, é não iterativo. Contudo, uma posterior expansão deste sistema, a fim de torná-lo conversacional ou iterativo não repercutirá sobre as subrotinas destinadas à geração da Lista do Display ou sobre o programa de controle e geração da imagem. Caso um novo tipo de informação seja criado na LD, será necessário apenas incluir no programa de controle o conjunto de instruções que manipulará esta nova informação, pois o desvio indexado desviará o processamento do programa segundo o valor da chave da informação que será manipulada.

Se, ao nível do display, forem conectados outros dispositivos tais como light pen, keyboard, etc, torna-se necessária uma subrotina que controle o lançamento deste nível, identificando qual dos dispositivos emitiu o sinal de interrupção e propiciando a execução do programa relativo àquele dispositivo.

Uma solução viável para a iteração homem-máquina, via teletipo, é prover o quadro de funções com uma subrotina que ao ser lançada salve o contexto do programa residente que propicia a interface monitor-operador, associado ao nível de interrupção do teletipo e o substitua pelo contexto do programa que analisará e executará as ordens recebidas por intermédio do teletipo. Uma determinada ordem suspenderá este programa, devolvendo o processamento à subrotina supra-citada que, após restaurar o contexto do pro-

grama residente, se desativará.

Em vista da atual configuração do sistema MITRA 15 não permitir a utilização da linguagem FORTRAN, as subrotinas apresentadas foram testadas utilizando-se a linguagem ASSEMBLER . Os parâmetros das subrotinas foram transmitidos utilizando-se os registros Acumulador, Extensão e Registro de Índice.

O algoritmo desenvolvido para a subrotina CIRCO foi testado utilizando-se a linguagem BASIC disponível no sistema MITRA 15 e a linguagem FORTRAN disponível no sistema 1130 da IBM.

As indisponibilidades do monitor de tempo real - MTR e do hardware do display impossibilitaram o teste do programa de controle e exibição da imagem, estando este programa, portanto, sujeito a apresentar fortuitos erros que facilmente poderão ser corrigidos.

A P Ê N D I C ELISTAGEM DA PROGRAMAÇÃO DOS ALGORITMOS

```

1          *LISTAGEM SUBROTINA DEFIG
2          CDS1  CDS
3          FIN
4          LDS1  LDS
5 0000      RES  2
6 0004 0000  DATA 0
7 0006 0000A PTARG DATA TARG
8 0008 02    DATA,1 2  *NUMERO DE ARGUMENTOS-N
9 0009 00    DATA,1 0
10 000A      TARG EQU $  *PONTEIRO P/ TABELA DE AR
11 000A      ENDER RES 2  *GUMENTOS
12 000E      IN  RES 2
13          REF  A3
14          REF  IMAGE
15 0012 0000A RI  DATA A3
16 0014 0000A IMAG DATA IMAGE
17          FIN
18          DEFIG LPS  LDS1
19          *TRANSFERENCIA DOS PARAMETROS
20 0000 2108  DAC  LDE  =3  *(4XN)
21 0002 0206  LDX  PTARG
22 0004 0000  CSV  M:MOVE
23 0006 6212  LDX  @RI
24 0008 A414  LEA  @IMAG,X *ENDEREÇO DA SUBFIG. OU
25 000A 710A  STA  @ENDER  *FIG. NA LISTA DO DISPLAY
26 000C F902  LDR  =2
27 000E 770A  ADM  @ENDER  *ENDEREÇO ABS. DA SUBF. OU FIG.
28 0010 000E  LDA  IN
29 0012 C502  BAZ  $+2  *DESVIA SE FOR SUBFIGURA
30 0014 C704  BRU  $+4  *DESVIA SE FOR FIG. PRINCIPAL
31          *GERAR PONTEIRO TIPO 3
32 0016 2000  LDA  =0
33 0018 2D05  LBL  =5
34 001A B114  STA  @IMAG,X
35 001C F204  ICX  =4
36 001E 7312  STX  @RI  *ATUALIZA RI
37 0020 F100  RTS
38          FIN  DAC
39          END
NSV  0  NB.ERR  0
%ASS2/

```

```

1          *LISTAGEM DA SUBROTINA FIM
2          CDS2  CDS
3          FIN
4          LDS2  LDS
5          RES  2
6          DATA 0
7          PTARG DATA TARG
8          DATA,1 1
9          DATA,1 0
10         TARG  EQU  $
11         ENDER RES  2
12         IN    RES  2
13         REF   A1
14         REF   A2
15         REF   A3
16         REF   IMAGE
17         C1    DATA A1
18         C2    DATA A2
19         RI    DATA A3
20         IMAG  DATA IMAGE
21         FIN
22         FIM   LPS  LDS2
23         *TRANSFERENCIA DOS PARAMETROS
24         DAC   LDE  =3
25         LDX   PTARG
26         CSV   M:MOVE
27         LDA   IN
28         BAZ   $+2          *DESVIA SE FOR SUBFIGURA
29         BRU   $+10         *DESVIA SE FOR FIG. PRINCIPAL
30         LDA   @C1
31         BCF   $+3          *DESVIA SE C1 0
32         LDA   @C2
33         BCT   $+6          *DESVIA SE C2 0
34         LDE   @C1
35         CNA
36         XAE
37         CNA
38         CLS   CINT1
39         LDA   =0
40         STA   @C1
41         STA   @C2
42         LDX   @RI
43         *GERAR APONTADOR TIPO 1
44         LBL   =1
45         STA   @IMAG,X
46         ICX   =2
47         LDA   @ENDER
48         STA   @IMAG,X
49         ICX   =2
50         STX   @RI          *ATUALIZA RI
51         RTS
52         FIN   DAC
53         END
NSV  0  NB.ERR  0
%ASS2/

```

```

1      *LISTAGEM SUBROTINA GEFIG
2      CDS4  CDS
3      FIN
4      LDS4  LDS
5      RES  2
6      DATA 0
7      PTARG DATA TARG
8      DATA,1 2
9      DATA,1 0
10     TARG  EQU  $
11     TAB   RES  2
12     CONT  RES  2
13     REF   A1
14     REF   A2
15     REF   A3
16     REF   IMAGE
17     C1    DATA A1
18     C2    DATA A2
19     RI    DATA A3
20     IMAG  DATA IMAGE
21     IND   DATA 0
22     SALVE RES  1
23     FIN
24     GEFIG LPS  LDS4
25     *TRANSFERENCIA DOS PARAMETROS
26     0000 2108  INIT  LDE  =8
27     0002 0206  LDX   PTARG
28     0004 0000  CSV   M:MOVE
29     *POSICIONAMENTO DO BIT DA LUMINOSIDADE
30     0006 021A  LDX   IND
31     0008 A00A  LDA   @TAB,X
32     000A F202  ICX   =2
33     000C C506  BAZ   $+6      *DESVIA SE O SEGMENTO NAO
34     000E 131A  STX   IND      *E LUMINOSO
35     0010 6216  LDX   @RI
36     0012 2004  LDA   =4
37     0014 B118  STA   @IMAG,X
38     *CALCULAR  $\Delta X$ 
39     0016 021A  LDX   IND
40     0018 A00A  LDA   @TAB,X
41     001A 6612  SUB   @C1
42     *ATUALIZAR C1
43     001C F102  XAE
44     001E A00A  LDA   @TAB,X
45     0020 7112  STA   @C1
46     *CALCULAR  $\Delta Y$ 
47     0022 F202  ICX   =2
48     0024 A00A  LDA   @TAB,X
49     0026 6614  SUB   @C2
50     *ATUALIZAR C2
51     0028 111C  STA   SALVE
52     002A A00A  LDA   @TAB,X
53     002C 7114  STA   @C2

```

```

54 002E 001C          LDA  SALVE
55 0030 F202          ICX  =2
56 0032 111A          STA  IND
57 0034 0000          CLS  CINT1
58                    *TESTE SE É O ULTIMO SEGMENTO
59 0036 000E          LDA  CONT
60 0038 2601          SUB  =1
61 003A 110E          STA  CONT
62 003C C502          BAZ  $+2
63 003E CF1C          BRU  $-28
64 0040 111A          STA  IND          *REPOE O CONTADOR IND A ZERO
65 0042 F100          RTS
66                    FIN  INIT
67                    END
NSV  0  NB.ERR      0
%ASS2/

```

```

1          *LISTAGEM SUBROTINA LIGSF
2          CDS3  CDS
3          FIN
4          LDS3  LDS
5 0000      RES  2
6 0004 0000  DATA 0
7 0006 0000A PTARG DATA TARG
8 0008 01    DATA,1 1
9 0009 00    DATA,1 0
10         000A TARG EQU 3
11 000A      ENDER RES 2
12          REF  A3
13          REF  IMAGE
14 000E 0000A RI    DATA A3
15 0010 0000A IMAG  DATA IMAGE
16          FIN
17          LIGSF LPS  LDS3
18          *TRANSFERENCIA DO PARAMETRO
19 0000 2104  CA    LDE  =4
20 0002 0206  LDX  PTARG
21 0004 0000  CSV  M:MOVE
22          *GERAR PRIMEIRA PALAVRA DO APONTADOR
23 0006 620E  LDX  @RI
24 0008 2000  LDA  =0
25 000A 2D02  LBL  =2
26 000C B110  STA  @IMAG,X
27          *ARMAZENAR O ENDEREÇO DA SUBFIG. NO APONT.
28 000E F202  ICX  =2
29 0010 600A  LDA  @ENDER
30 0012 B110  STA  @IMAG,X
31 0014 F202  ICX  =2
32 0016 730E  STX  @RI          *ATUALIZA RI
33 0018 F100  RTS
34          FIN  CA
35          END
NSV  0  NB.ERR  0
%ASS2/

```



```

1      *LISTAGEM SUBROTINA CINT1
2      CDS5  CDS
3      FIN
4      LDS5  LDS
5      0000      RES  2
6      0004      SALVE RES  1
7      0006 0100  TESTE DATA 256
8      REF  A3
9      REF  IMAGE
10     0003 0000A  RI  DATA A3
11     000A 0000A  IMAG DATA IMAGE
12     000C      PX  RES,1 1
13     000D      PY  RES,1 1
14           000C  PKY  EQU  PX
15      FIN
16     CIJTI LPS  LDS5
17     *POSICIONAMENTO DO BIT DO SINAL DE  X
18     0000 6203  INI  LDX  @RI
19     0002 C402      BAN  $+2      *DESVIA SE  X<0
20     0004 C707      BRU  $+7      *DESVIA SE  X>=0
21     0006 F11C      CNA                      *CALCULO DO VALOR ABSOLUTO
22     0008 1104      STA  SALVE
23     000A 2001      LDA  =1
24     000C A70A      IOR  @IMAG,X
25     000E B10A      STA  @IMAG,X
26     0010 0004      LDA  SALVE
27     *POSICIONAMENTO DO BIT DO SINAL DE  Y
28     0012 F102      XAE
29     0014 C402      BAN  $+2
30     0016 C707      BRU  $+7
31     0013 F11C      CNA
32     001A 1104      STA  SALVE
33     001C 2002      LDA  =2
34     001E A70A      IOR  @IMAG,X
35     0020 B10A      STA  @IMAG,X
36     0022 0004      LDA  SALVE
37     0024 0306      CMP  TESTE      *COMPARA  Y COM 256
38     0026 C60B      BGE  $+11      *DESVIA SE  Y>=256
39     0028 F102      XAE
40     002A 0306      CMP  TESTE      *COMPARA  X COM 256
41     002C C607      BGE  $+7      *DESVIA SE  X>=256
42     002E F103      XAA
43     0030 F116      AIE                      *REUNE  X E  Y NO ACUMULADOR
44     0032 F202      ICX  =2
45     0034 B10A      STA  @IMAG,X *ARMAZENA-OS NA LISTA DO
46     0036 C71A      BRU  $+26      *DISPLAY
47     0033 F102      XAE
48     003A F0C1      SRLS =1      *DIVIDE  Y POR 2
49     003C C305      BCF  $+5
50     003E 1104      STA  SALVE
51     0040 2001      LDA  =1
52     0042 150D      SBR  PY
53     0044 0004      LDA  SALVE

```

54 0046 F103
 55 0048 F001
 56 004A C305
 57 004C 1104
 58 004E 2001
 59 0050 150C
 60 0052 0004
 61 0054 F103
 62 0056 F116
 63 0058 A10A
 64 005A F202
 65 005C B10A
 66 005E F202
 67 0060 050C
 68 0062 B60A
 69 0064 F204
 70 0066 7303
 71 0068 2000
 72 006A 110C
 73 006C F100
 74
 75

XAA
 SRLS =1
 BCF \$+5
 STA SALVE
 LDA =1
 SBR PX
 LDA SALVE
 XAA
 AIE
 LDE @IMAG,X
 ICX =2
 STA @IMAG,X
 ICX =2
 ADD PXY
 DST @IMAG,X *DUPLICA A INFORMACAO
 ICX =4 *TIPO 1
 STX @RI
 LDA =0
 STA PXY
 RTS
 FIN INI
 EJD

VS' 0 VB.ERR 0
 %ASS2/

```

C      LISTAGEM SUBROTINA CIRC0
      SUBROUTINE CIRC0(ENDD1,RA)
      INTEGER X1,Y1,RA1,IND,END2
      RA1=RA
      CALL DEFIG(ENDD2,0)
      ENDD1=END2
      IF (RA-12)10,10,15
10    CALL CRIMD(RA1)
      GO TO 45
15    NSEG=RA/4+0.5
      IC1=INT(RA)
      IC2=0
      ANG1=6.28/(8*NSEG)
      ANG2=ANG1
      L=NSEG-1
      IND=1
      DO 30 J=1,L
      IX=INT(RA*COS(ANG2))
      X=FLOAT(IX)
      IY=INT(SQRT(RA*RA-X*X))
      X1=-(IX-IC1)
      Y1=IY-IC2
      IF(X1)25,20,25
20    IF(Y1)25,30,25
25    IC1=IX
      IC2=IY
      CALL CIRC(X1,Y1,RA1,IND)
      ANG2=ANG2+ANG1
30    IND=0
      IX=INT(RA/1.414+0.5)
      X1=-(IX-IC1)
      Y1=IX-IC2
      IF(X1)40,35,40
35    IF(Y1)40,45,40
40    CALL CIRC(X1,Y1,RA1,IND)
45    CALL FIM(ENDD2,0)
      RETURN
      END

```

```

1      *LISTAGEM SUBROTINA CRIMD
2      CDS1  CDS
3      FIN
4      LDS3  LDS
5      0000      RES  2
6      0004 0000      DATA 0
7      0006 0000A    PTARG DATA TARG
8      0008      01      DATA,1 1
9      0009      00      DATA,1 0
10     000A    TARG EQU  $
11     000A    RA      RES  2
12     000E 5000      CIR1  DATA &5000
13     0010      PONT  RES  1
14     0012 1540      CIR2  DATA &1540
15     0014      CONT  RES  1
16     0016 1A40      CIR3  DATA &1A40
17     0018      CON   RES  1
18     001A 1694      CIR4  DATA &1694
19     001C 0000      DATA &0000
20     001E 1659      CIR5  DATA &1659
21     0020 4000      DATA &4000
22     0022 2659      CIR6  DATA &2659
23     0024 8000      DATA &8000
24     0026 17AD      CIR7  DATA &17AD
25     0028 4000      DATA &4000
26     002A 17FD      CIR8  DATA &17FD
27     002C 4000      DATA &4000
28     002E 2AA5      CIR9  DATA &2AA5
29     0030 9980      DATA &9980
30     0032 27AA      CIR10 DATA &27AA
31     0034 D800      DATA &D800
32     0036 27BE      CIR11 DATA &27BE
33     0038 D800      DATA &D800
34     003A 27B5      CIR12 DATA &27B5
35     003C ED80      DATA &ED80
36     003E 000E    PTABC DATA CIR1  *APONTADOR P/ TABELA DE PONTEIROS
37     REF A3      **DAS EXTRUTURAS COMPACTADAS
38     REF IMAGE
39     0040 0000A    RI      DATA A3
40     0042 0000A    IMAG   DATA IMAGE
41     FIN
42     CRIMD LPS LDS8
43     *TRANSFERENCIA DO PARAMETRO
44     0000 0206    INI     LDX PTARG
45     0002 2104      LDE   =4
46     0004 0000      CSV   M:MOVE
47     *ARMAZENAR NA LISTA DO DISPLAY O SEGMENTO
48     *QUE DESLOCA O RAI0 LUMINOSO P/ A PERIFE-
49     *RIA DO CIRCULO
50     0006 6240      LDX   0RI
51     0008 F202      ICX   =2
52     000A 600A      LDA   @RA
53     000C F108      XAA

```

```

54 000E B142          STA  @IMAG,X
55 0010 F202          ICX  =2
56 0012 7340          STX  @RI
57                    *GUARDAR O ENDERECO DA 1A. PALAV. DA INF. TIPO 4
58 0014 1314          STX  CONT
59                    *COLETAR O ENDERECO DA TABELA COMPACTADA
60                    *DO RAI0 FORNECIDO
61 0016 600A          LDA  @RA
62 0018 2601          SUB  =1
63 001A 2C04          MUL  =4
64 001C F104          XAX
65 001E A13E          LA   LDE  @PTABC,X      *COLETA UMA PALAVRA DA ESTRU-
66 0020 1310          STX  PONT              **TURA COMPACTADA
67 0022 2000          LDA  =0
68 0024 1118          STA  CON
69 0026 6240          LDX  @RI
70                    *DECODIFICAR 1 PAR ( X, Y)
71 0028 F062          ACI   SLCD =2
72 002A F006          SLLS =6
73 002C F062          SLCD =2
74 002E C500A         BAZ  SALT
75 0030 F202          ICX  =2
76 0032 B142          STA  @IMAG,X
77 0034 2001          LDA  =1
78 0036 1718          ADM  CON
79 0038 2B04          CMP  =4
80 003A C603          BGE  $+3
81 003C 2000          LDA  =0
82 003E CF0B          BRU  ACI              *DECODIFICAR NOVO PAR
83 0040 7340          STX  @RI
84 0042 0210          LDX  PONT
85 0044 F202          ICX  =2
86 0046 CF14          BRU  LA              *COLETAR NOVA PALAVRA
87                    *ATUALIZAR RI E CONSTRUIR A 1A. PALAVRA DA
88                    *INFOTMACAO TIPO 4
89 0048 F905          SALT LDR  =5
90 004A F202          ICX  =2
91 004C 7340          STX  @RI
92 004E 0614          SUB  CONT
93 0050 0214          LDX  CONT
94 0052 2D06          LBL  =6
95 0054 B142          STA  @IMAG,X
96 0056 F100          RTS
97                    FIN  INI
98                    END

```

NSV 0 NB.ERR 0

```

1      *LISTAGEM SUBROTINA CIRC
2      CDS1   CDS
3          FIN
4      LDS9   LDS
5      0000   RES   2
6      0004 0000   DATA 0
7      0006 0000A PTARG DATA TARG
8      0008   04   DATA,1 4
9      0009   00   DATA,1 0
10     000A   TARG EQU   $
11     000A   X     RES   2
12     000E   Y     RES   2
13     0012   Z     RES   2
14     0016   T     RES   2
15     001A   CONT  RES   1
16         REF   A3
17         REF  IMAGE
18     001C 0000A RI     DATA A3
19     001E 0000A IMAG  DATA IMAGE
20         FIN
21     *
22     CIRC   LPS   LDS9
23     *TRANSFERENCIA DOS PARAMETROS
24     0000 2110 LI     LDE   =16
25     0002 0206     LDX   PTARG
26     0004 0000     CSV   M:MOVE
27     0006 621C     LDX   @RI
28     *TESTAR SE E A PRIMEIRA VEZ QUE E CHAMADA
29     0008 6016     LDA   @T
30     000A C500A     BAZ   VA
31     000C F202     ICX   =2
32     000E 6012     LDA   03
33     0010 F108     XAA
34     0012 B11E     STA   @IMAG,X
35     *CONSTRUIR PRIMEIRA PALAVRA INFORMACAO TIPO 3
36     0014 F202     ICX   =2
37     0016 A41E     LEA   @IMAG,X
38     0018 111A     STA   CONT
39     001A 2000     LDA   =0
40     001C 2E03     LBR   =3
41     001E B11E     STA   @IMAG,X
42     0020 F202     ICX   =2
43     *ARMAZENAR UM PAR ( X , Y ) NA LD
44     0022 720E     VA     STE   @Y
45     0024 600A     LDA   @X
46     0026 F103     XAA
47     0028 F116     AIE
48     002A B11E     STA   @IMAG,X
49     002C F202     ICX   =2
50     002E 731C     STX   @RI
51     *INCREMENTAR O CONTADOR
52     0030 021A     LDX   CONT
53     0032 2002     LDA   =2
54     0034 B71E     ADM   @IMAG,X
55     0036 F100     RTS
56         FIN   LI
57         EJD

```

NSV 0 VB.ERR 0

%%EOD?

```

1      *LISTAGEM SUBROTINA GEMEN
2      CDS7  CDS
3      FIN
4      LDS7  LDS
5      RES  2
6      DATA 0
7      PTARG DATA TARG
8      DATA,1 4
9      DATA,1 0
10     TARG  EQU  $
11     TAB   RES  2
12     X     RES  2
13     Y     RES  2
14     NUM   RES  2
15     TESTE DATA 256
16     IND   DATA 0
17     CONT  RES  1
18     REF  A3
19     REF  IMAGE
20     RI    DATA A3
21     IMAG  DATA IMAGE
22     FIN
23     GEMEN LPS  LDS7
24     *TRANSFERENCIA DOS PARAMETROS
25     LAC   LDE  =16
26     LDX   PTARG
27     CSV   M:MOVE
28     *ARMAZENAR 1A. INFOR. TIPO 1 COM  X=0 E  Y=0
29     LDX   @RI
30     ICX   =4
31     STX   @RI
32     *CONSTRUIR 2A. INFOR. TIPO 1 COM  X=@X E  Y=@Y
33     LDA   @X
34     LDE   @Y
35     CLS   CINT1
36     LDA   @NUM
37     STA   CONT
38     *ARMAZENAR 2 CARACTERES NA LISTA DO DISPLAY
39     LA    LDX  IND
40     LDA   @TAB,X
41     XAA
42     XAE
43     ICX   =2
44     LDA   @TAB,X
45     AIE
46     ICX   =2
47     STX   IND
48     LDX   @RI
49     STA   @IMAG,X
50     ICX   =2
51     STX   @RI
52     LDA   CONT
53     SUB   =1

```

54	0034	C503	BAZ	\$+3
55	0036	111E	STA	CONT
56	0038	CF11	BRU	LA
57	003A	111C	STA	IND
58	003C	F100	RTS	
59			FIN	LAC
60			END	
NSV	0	NB.ERR	0	
%ASS2/				


```

1          *LISTAGEM SUBROTINA LPCEI
2          CDS1  CDS
3          FIN
4          LDS6  LDS
5          RES  2
6          DATA 0
7          PTARG DATA TARG
8          DATA,1 1
9          DATA,1 0
10         TARG EQU  $
11         ENDER RES  2
12         ZC     RES  1
13         DEF  A1
14         DEF  A2
15         DEF  A3
16         DEF  IMAGE
17         A1     RES  1
18         A2     RES  1
19         A3     RES  1
20         IMAGE RES  400
21         FIN
22         LPCEI LPS  LDS6
23         COM   LDE   =4
24         LDX   PTARG
25         CSV M:MOVE
26         *ARMAZENAR ENDER. FIG. PRINCIPAL EM ZC
27         LDA  =6
28         STA  ZC
29         LDA  @ZC
30         STA  ZC
31         LDA  @ENDER
32         STA  @ZC
33         *DESVIAR PARA O PROGRAMA DE CONTROLE
34         LDX  =2
35         LDA  =6
36         CSV  M:IT
37         *LACO DE ESPERA DA INTERRUPCAO DO DISPLAY
38         BRN  $
39         RTS
40         FIJ  COM
41         EJD

```

```

NSV  0  VB.ERR  0
%ASS2/

```

```

1      *LISTAGEM DO PROGRAMA DE CONTROLE E EXIBICAO
2      **DA IMAGEM
3      CODI   CDS
4 0000      COM   RES   16
5          0006   ZC   EQU   COM+6   *APONTA P/ A ZONA CUMUM DO SISTEA
6          0006   RED   EQU   COM+6   * " " ,COM RX=2, PARA A 2A. PALA
7 0020      R1   RES   1           **VRA DA ZC QUE SERA USADA COMO /
8 0022      R2   RES   1           **REGISTRO DO DISPLAY
9 0024      CONT  RES   1
10         FIN
11         L2     LDS
12 0000      RES   2
13 0004      DIS   RES   1
14 0006      A1    RES   1
15          0007   A2    EQU   A1+1
16 0008      CL    RES,1 1
17 0009      00A   ITAB  DATA,1 IT
18 000A      06    IT    DATA,1 6   *TABELA DAS CONDICOOES DOS SEGME
19 000B      07    DATA,1 7   **TOS QUE DEFINEM OS CIRCULOS
20 000C      05    DATA,1 5
21 000D      04    DATA,1 4
22 000E      RX    RES   1
23         FIN
24         EXEC   LPS   L2
25 0000 1604      EXE   DST   DIS
26 0002 4024      LDA   #CONT
27 0004 1503      SBR   CL
28 0006 0E06      ACI   LBR   A1
29 0008 C504      BAZ   $+4
30 000A 6004      LDA   @DIS
31 000C F108      XAA
32 000E C702      BRU   $+2
33 0010 6004      LDA   @DIS
34 0012 5122      STA   #R2
35 0014 AE09      LBR   @ITAB,X
36 0016 5120      STA   #R1
37 0018 130E      STX   RX
38 001A F401      DIT
39 001C 5020      DLD   #R1
40 001E 2202      LDX   =2
41 0020 9606      DST   @#RED,X   *TRANSFERE A INFORMACAO AOS
42 0022 020E      LDX   RX       **REGISTROS DO DISPLAY
43 0024 0E03      LBR   CL
44 0026 2601      SUB   =1
45 0028 C500A     BAZ   LUG
46 002A 1508      SBR   CL
47 002C 0E07      LBR   A2
48 002E C505      BAZ   $+5
49 0030 0004      LDA   DIS
50 0032 2602      SUB   =2
51 0034 1104      STA   DIS
52 0036 CF13      BRU   ACI
53 0038 2002      LDA   =2

```

54	003A	1704	ADM	DIS	
55	003C	CF1B	BRU	ACI	
56	003E	1004	LUG	DLD	DIS
57	0040	F100	RTS		
58			FIN	EXE	
59			*		
60			L1	LDS	
61	0000		BG	RES	1
62	0002		GI	RES	1
63			FIN		
64			EXIBA	LPS	L1
65	0000	F902	JO	LDR	=2
66	0002	1100		STA	BG
67			*COLETAR O ENDEREÇO DA FIG PRINCIPAL NA ZC		
68	0004	2200		LDX	=0
69	0006	8006		LDA	@#ZC,X
70	0008	0600		SUB	BG
71	000A	1102		STA	GI
72	000C	6E02	VOLTE	LBR	@GI
73	000E	C101		BRX	\$+1
74	0010	C700A		BRU	DESV0
75	0012	C700A		BRU	DESV1
76	0014	C700A		BRU	DESV2
77	0016	C700A		BRU	DESV3
78	0018	C700A		BRU	DESV4
79	001A	C700A		BRU	DESV5
80	001C	C700A		BRU	DESV6
81			*PROCESSAMENTO INFORMACAO TIPO 1		
82	001E	7002	DESV0	DLD	@GI
83	0020	5620		DST	#R1
84	0022	F401		DIT	
85	0024	5020		DLD	#R1
86	0026	2202		LDX	=2
87	0028	9606		DST	@#RED,X
88	002A	2004		LDA	=4
89	002C	1702		ADM	GI
90	002E	CF11		BRU	VOLTE
91			*PROCESSAMENTO APONTADOR TIPO 1		
92	0030	2002	DESV1	LDA	=2
93	0032	1702		ADM	GI
94	0034	6002		LDA	@GI
95	0036	0600		SUB	BG
96	0038	1102		STA	GI
97	003A	CF17		BRU	VOLTE
98			*PROCESSAMENTO APONTADOR TIPO 2		
99	003C	0202	DESV2	LDX	GI
100	003E	F204		ICX	=4
101	0040	2002		LDA	=2
102	0042	1702		ADM	GI
103	0044	6002		LDA	@GI
104	0046	0600		SUB	BG
105	0048	1102		STA	GI
106	004A	2002		LDA	=2

107	004C	1702	ADM	GI
108	004E	7302	STX	@GI
109	0050	2002	LDA	=2
110	0052	1702	ADM	GI
111	0054	CF24	BRU	VOLTE
112			*PROCESSAMENTO INFORMACAO TIPO 2	
113	0056	2001	DESV3	LDA =1
114	0058	1702	ADM	GI
115	005A	6002	LDA	@GI
116	005C	5124	STA	#CONT
117	005E	2001	LDA	=1
118	0060	1702	ADM	GI
119	0062	2200	LDX	=0
120	0064	F102	XAE	
121	0066	2000	LA	LDA =0
122	0068	0000	CLS	EXEC
123	006A	2E01	LBR	=1
124	006C	2D01	LBL	=1
125	006E	0000	CLS	EXEC
126	0070	F201	ICX	=1
127	0072	2E01	LBR	=1
128	0074	0000	CLS	EXEC
129	0076	2000	LDA	=0
130	0078	2D01	LBL	=1
131	007A	0000	CLS	EXEC
132	007C	F104	XAX	
133	007E	2603	SUB	=3
134	0080	C503	BAZ	\$+3
135	0082	2202	LDX	=2
136	0084	CF0F	BRU	LA
137	0086	4024	DES	LDA #CONT
138	0088	1702	ADM	GI
139	008A	CF3F	BRU	VOLTE
140			*PROCESSAMENTO INFORMACAO TIPO 3	
141	008C	2001	DESV4	LDA =1
142	008E	1702	ADM	GI
143	0090	6002	LDA	@GI
144	0092	5124	STA	#CONT
145	0094	2001	LDA	=1
146	0096	1702	ADM	GI
147	0098	6E02	VOL	LBR @GI
148	009A	2D80	LBL	=123
149	009C	5120	STA	#R1
150	009E	6002	LDA	@GI
151	00A0	2D80	LBL	=123
152	00A2	5122	STA	#R2
153	00A4	F401	DIT	
154	00A6	5020	DLD	#R1
155	00A8	2202	LDX	=2
156	00AA	9606	DST	@#RED,X
157	00AC	2002	LDA	=2
158	00AE	1702	ADM	GI
159	00B0	4024	LDA	#CONT

160	00B2	2601	SUB	=1	
161	00B4	C503	BAZ	\$+3	
162	00B6	5124	STA	#CONT	
163	00B8	CF10	BRU	VOL	
164	00BA	CF57	BRU	VOLTE	
165			*PROCESSAMENTO APONTADOR TIPO 3		
166	00BC	2002	DESV5	LDA	=2
167	00BE	1702	ADM	GI	
168	00C0	6002	LDA	@GI	
169	00C2	1102	STA	GI	
170	00C4	CF5C	BRU	VOLTE	
171			*PROCESSAMENTO INFORMACAO TIPO 4		
172	00C6	2001	DESV6	LDA	=1
173	00C8	1702	ADM	GI	
174	00CA	6002	LDA	@GI	
175	00CC	5124	STA	#CONT	
176	00CE	2001	LDA	=1	
177	00D0	1702	ADM	GI	
178	00D2	2200	LDX	=0	
179	00D4	F102	XAE		
180	00D6	2000	LDA	=0	
181	00D8	0000	CLS	EXEC	
182	00DA	F201	ICX	=1	
183	00DC	2001	LDA	=1	
184	00DE	0000	CLS	EXEC	
185	00E0	F201	ICX	=1	
186	00E2	2000	LDA	=0	
187	00E4	0000	CLS	EXEC	
188	00E6	F201	ICX	=1	
189	00E8	2001	LDA	=1	
190	00EA	0000	CLS	EXEC	
191	00EC	CF33	BRU	DES	
192			FIN	JO	
193			END	EXIBA	
VSV 2 NB.ERR 2					
%ASS2/					

BIBLIOGRAFIA

- 1 - PARLOW, R. D. and GREEN, R. E. -Advanced Computer Graphics. Plenum Press-1971
- 2 - PRINCE, M. D. -Interactive Graphics for Computer Aided Design. Addison-Wesley Publishing Company-1971
- 3 - DESMONDE, W. H. -A Conversational Graphic Data Processing System: The IBM 1130/2250. Prentice-Hall Inc.-1969
- 4 - CII. -Manuel de Présentation: MITRA 15. Compagnie Internationale pour L'Informatique-1972